

icy-url:http://www.shoutcast.com\r\n

icy-irc:%23shoutcast\r\n

icy-pub:1\r\n
icy-br:56\r\n

icy-icq:0\r\n

icy-aim:N%2FA\r\n
\r\n

Then The source will begin sending the mp3 encoded stream

- * icy-name is the name of the stations
- * icy-genre is the genre that the station resides in
- * icy-pub is basically a switch to either allow the server to publish itself in the directory or not (1 meaning yes and 0 meaning no)
- * icy-br is the bitrate of the stream
- * icy-url is the homepage for the stream
- * icy-irc is yp shoutcast specific (used for contact information)
- * icy-icg is yp shoutcast specific (used for contact information)
- * icy-aim is yp shoutcast specific (used for contact information)

You can also pass this optional data:

CODE: SELECT ALL

content-type: mime/type\r\n

icy-reset: 1\r\n
icy-prebuffer: ??\r\n

- * content-type is the data type to expect from this stream. (HTTP spec header)
- * icy-reset tells the server whether it should clear out the buffer. (necessary for NSV/NSA streams.
- * icy-prebuffer, we aren't quite certain what this is for, how to use it or even whether it works, but it exists.

The optional params are not necessarily passed to the client, content-type is of course but as for the others it is not clear.

This is just a simple walk through of how the source communicates with the server. No other information is passed on this port as far as I am aware.

Title streaming from source to server

This is a simple one, the server receives the title of the song and the URL of the page simply by having the source make the URL call

CODE: SELECT ALL

http://www.host.com:portnumber/admin.cgi?pass=Server%20Password&mode=updinfo

When this gets called by the source or a browser even, the title of the song changes in the clients which support shoutcast style title streaming. This communication always happens on the public port (by default 8000) never on the service port as it is used for strictly sending the stream to the server.

You also must make sure that when you make your HTTP calls that it comes from a browser or program that specifies the User-Agent: header as Mozilla.

Client to Server

The Client to Server communication is handle in a similar fashion to the way that a browser communicates with a webpage server. This is known as the HTTP protocol. However SHOUTcast and icecast do not handle in exactly the same manner, the headers are different. I have yet to pin point exactly what is so different. I think it may have something to do with the notification error, as it is HTTP/1.0 200 OK on all web servers using HTTP, this may confuse some clients and causes the headers to not exist.

1. The client connects to the server and sends information about itself, if it can handle title streaming it sends and extra field like so:

CODE: SELECT ALL

icy-metadata:1\r\n

In addition to the normal headers sent. This tag signifies that the client has the ability to stream the title streaming tags from the stream, therefore the server will send the extra title information, if this were not possible, some clients would hiccup when the title information is sent

2. The server then responds with

CODE: SELECT ALL

icy-genre:Unknown Genre\r\n (what genre the server falls under)

 $icy-url:http://www.shoutcast.com\r\n$ (homepage for the server)

Content-Type:audio/mpeg\r\n (Content type of the stream to follow)

icy-pub:1\r\n (whether the server is public or not)

icy-br:56\r\n (bitrate of the server)

icy-metaint:8192\r\n (if icy-metadata:1 was signified this was shown I will

\r\n (end of header)

3. At this point the server begins sending the audio data.

SHOUTcast Meta Title Streaming

Earlier we discussed how the server gets the title of the song from the source, but we didn't quite get into how the client gets the title of the song.

When the client signifies that it is title streaming compatible, the shoutcast server adds an extra header tag set like so

CODE: SELECT ALL

icy-metaint:8192\r\n

this tells the client exactly how many bytes of data to read out of the stream before it can expect the beginning of the Meta-Data (which is where the title is stored) It also always starts counting at the beginning of the stream (not the header)

After this the client then reads 1 byte, this byte tells the client how large the Meta-Data Tag is divided by 16, so if the byte was 4 then the client would know that the meta-data tag was 64 bytes long. But, you ask, not all titles are going to equal 64 bytes or 48 bytes etc...? Well the simple answer is that SHOUTcast places blanks or "\0" in the unused space until it equals the length, after that is read, then it is back to the mpeg data to start the process all over again.

Pretty simple huh?

In Closing

I am sure that this technology will change, and I will try my best to keep this article up to the specs as I know them. If you find anything incorrect in this article, or any oversights, then please email me

Feel free to leave a comment if you feel that I have missed something. Do not reply with questions. Questions should go in the Audio Streaming forum. Questions will be split from this thread and moved to appropriate forums.

Last edited by Jay on Mon Jan 16, 2006 9:41 am, edited 6 times in total.



Passing the mp3 binary.

Wed Oct 27, 2004 11:04 am

Lets say that I don't want to bother with a DSP. What if I knew what mp3 i wanted to pass to the server but without the need of opening winamp, loading the dsp, and running the file.

I assume the following process:

Make a script/program, that loads into memory the mp3 that needs to be passed. Lame, or mp3pro, encode then pass the necessary informationt to the server in order to initiate a transfer. When the server returns icy 200 (ok) then what happens? How would I pass the mp3 to the server, I assume a socket connection but is there a specific method?

Thanks for the little lecture, hope I understood.

-Testi



ThuGie



YP Protocol

Fri Dec 31, 2004 5:25 am

I downloaded today the new shoutcast server and i was thinking i'm gone scan what is sending to the yp so i did and this is what i got not sure of anything changed.

First what you need to send

GET /addsrv?v=1&br=24&p=8000&m=3&t=Station

Name&g=Genre&url=http://shoutcast.com&content=audio/mpeg&irc=#shoutcast&icq=0&aim

=N/A HTTP/1.0

Host: yp.shoutcast.com

Accept: */*

this is what you send 😀

ok v=unknown to me sorry

br=bitrate

p=port

m=max users i gues

t=station name

g=genre

url=website url

content=type of stream

irc=irc channel

aim= gues 🙂



it always sends this or it got updated (5):

HTTP/1.1 200 OK

Date: Fri, 31 Dec 2004 11:07:27 GMT Server: Apache/1.3.33 (Unix) PHP/4.3.10

X-Powered-By: PHP/4.3.10

Connection: close

Content-Type: shoutcast/crapola

oke when you send it but did not enter a station name you get this error:

icy-response: nak

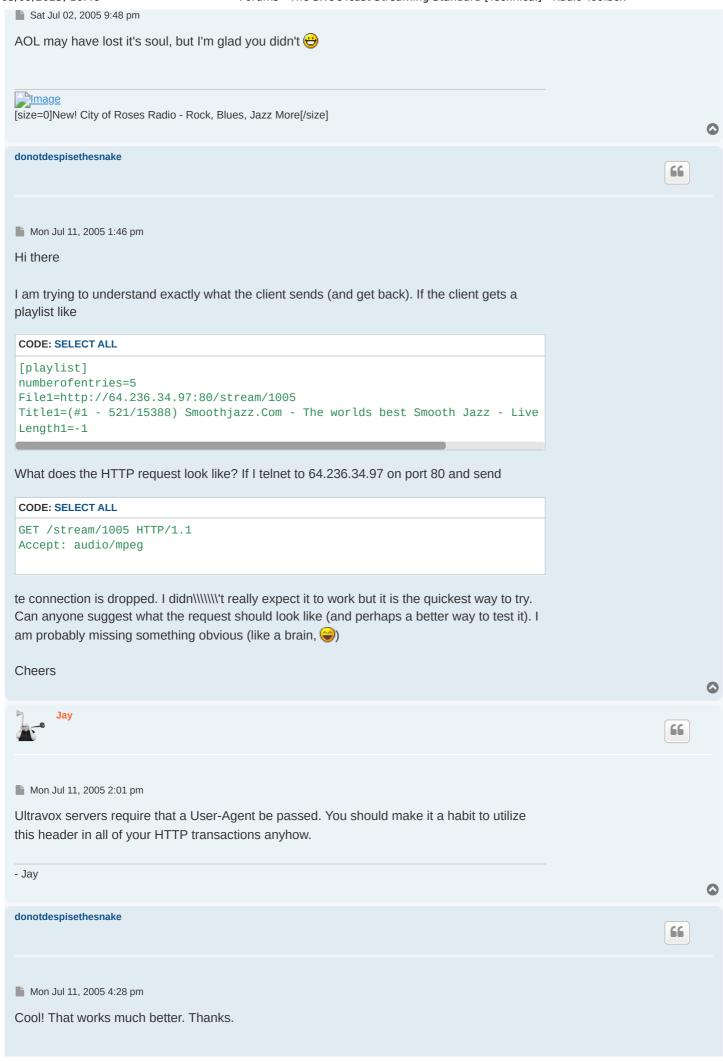
icy-error: please give your station a name (in the dsp plugin YP tab) oke when you did not send the genre: icy-response: nak icy-error: please identify this station's genre (in the dsp plugin YP tab) icy-response: nak = failed adding i gues when sending and succes. icy-response: ack icy-id: 75568272 icy-tchfrq: 3 icy-response: ack = succes i gues icy-id: ? = the id on the yp also used to remove yourself icy-tchfrq: 3 = unknown sorry oke now to remove yourself: GET /remsrv?id=?&p=8000 HTTP/1.0 Host: yp.shoutcast.com Accept: */* id=the id you got when succes adding. p=port of stream. when succes removing you get: icy-response: ack this is all i got :p i hope you can use it i dont think so but still :p. i was bored 😃 theres more i gues but this only took me 30sec to scan so 🖰. Jay 66 Mon Jan 31, 2005 11:45 pm There is definately alot more to it and there are certain restrictions as well. I will not publish our knowledge of their yp due to security concerns (hacking the directory is a big problem at shoutcast.com due to the heavy traffic shoutcast.com recieves). - Jay **Andrew** 66 Mon Feb 07, 2005 2:19 pm icy-tchfrq: 3 means that shoutcast will send updates to the yp.shoutcast.com every 3 minutes. As the yp sends this back they can lengthen or shorten the update time per server if they so wish. icy-id: xxxxx

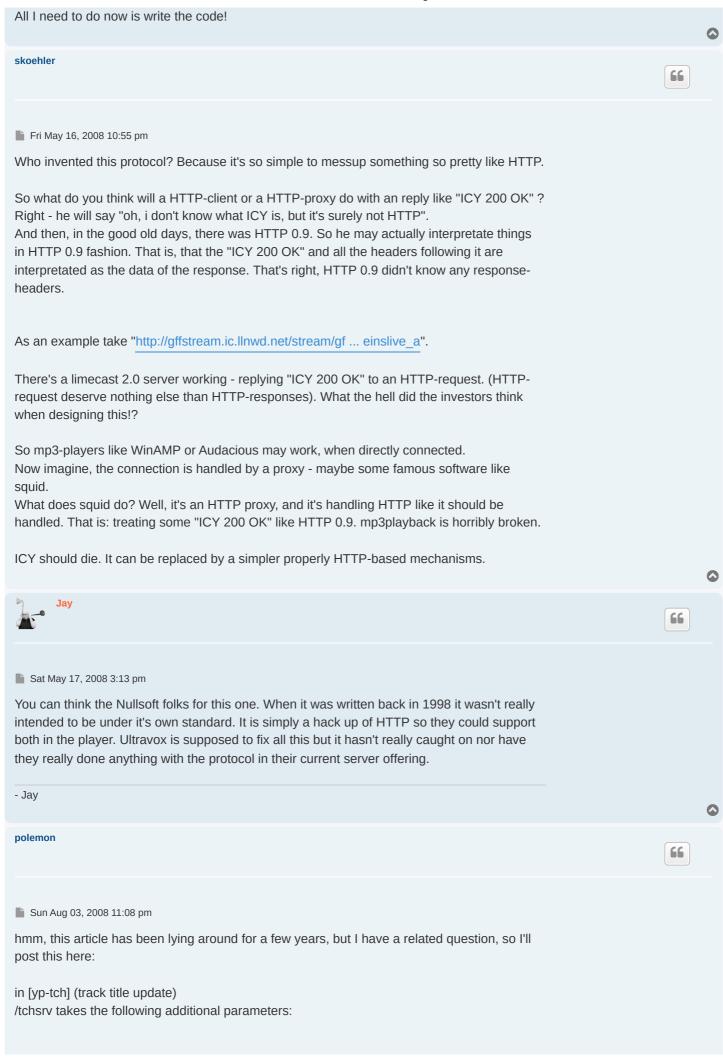
05/09/2023, 16:48 Forums - The SHOUTcast Streaming Standard [Technical] - Radio Toolbox is a unique Id within the shoutcast directory for the station entry. icy-backup: can also be sent back ... for some reason ... 😉 Andrew djspinnercee 66 **Excellent Info! ■** Wed Jun 29, 2005 12:18 pm I have been playing with this too.... a project I\'ve been playing with is a alternate directory app that behaves like the SHOUTcast directory.... There is some info that I can add that may be useful: I think that the icy-backup: will specify an DNAS id or IP that is used to help create clusters -so when this is sent to a DNAS it will \"know\" that backups exist to it can redirect when full. So far, I have found three commands that the DNAS sends to the YP: In forming its request, the important HTTP header for all is: Content-Type: shoutcast/crapola(0d)(0a)(0d)(0a) The DNAS always initiates (establishes) the connections to the YP. Then the icy-* things are in the content part of the request.... [yp-add] GET /addsrv? v=1 (don\'t know this, maybe 1=public, 0=private) br=bitrate p=port m=DNAS MaxListeners t=stream title g=genre url=URL irc= icq= aim= for this OP, the \"directory checker\" is called while the connection is still active -- it will try to extablish both a \"listen\" connection and a station page connection to determine accessability, validity, and I think to also get the current track info. if the directory checker returns an \"all good,\" it will finish this request with: icy-response: ack(0a) icy-id: nnnnnnn(0a) icy-tchfrq: 3(0a)and close the connection. [yp-rem] is as above.... GET /remsrv?

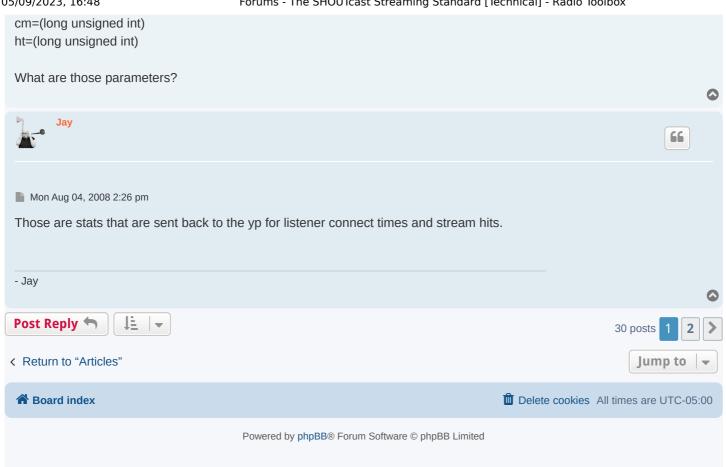
id=nnnnnnn

p=port

...the YP response body will only contain: icy-response: ack(0a) [yp-tch] Track title update... GET /tchsrv? id=nnnnnnn p=port li=current listeners alt=average listen time ct=now playing track title The YP responds: icy-tchfrq: 5(0a) icy-response: ack(0a) Errors are returned by the YP as: icy-response: nak(0a) icy-error: ...the appropriate message ...(0a) I assume that a NAK error will tell the DNAS to retry the failed OP at some other time. **** I have a simple YP sim running to test this.... because the DNAS does not allow config of the target YP (IP) and only the port (YPort=), I create a HOSTS file entry for yp.shoutcast.com nnn.nnn.nnn yp.shoutcast.com nnn.nnn.nnn is the raw IP found from a ping or tracert to www411.servehttp.com (the IP is dynamic, so it can change) in the DNAS config, the YPort=11427 Feel free to try it on your own --- so far the interaction is all that is going on, so it always returns ack -- I\m working on the database, and the directorry checker with intentions of being able to build a directory.... I\'m looking for feedback, or additional info that may be helpful.... Jav 66 Wed Jun 29, 2005 4:12 pm If you are interested in building a directory you might want to watch http://sourceforge.net/projects/yp-php/ I will be publishing my version of an Icecast2 directory there which will be able to list both Steamcast and Icecast2 streams. - Jay webranger 66







Terms of Service | ♠ Privacy Policy | ☑ Contact Us | 🤟 Follow Us © Radio Toolbox, LLC • 2006–2023