

NAME

rtmpgw - RTMP streaming media gateway

SYNOPSIS

```
rtmpgw [-r url] [-n hostname] [-c port] [-l protocol] [-S host:port] [-a app] [-t tcUrl] [-p pageUrl]
[-s swfUrl] [-f flashVer] [-u auth] [-C conndata] [-y playpath] [-v] [-d subscription] [-e] [-k skip]
[-A start] [-B stop] [-b buffer] [-m timeout] [-T key] [-j JSON] [-w swfHash] [-x swfSize] [-W swfUrl]
[-X swfAge] [-D address] [-g port] [-q] [-V] [-z]
rtmpgw -h
```

DESCRIPTION

rtmpgw is a server for streaming media content from RTMP out to HTTP.

rtmpgw listens for HTTP requests that specify RTMP stream parameters and then returns the RTMP data in the HTTP response. The only valid HTTP request is "GET /" but additional options can be provided in URL-encoded fashion. Options specified on the command line will be used as defaults, which can be overridden by options in the HTTP request.

OPTIONS**Network Parameters**

These options define how to connect to the media server.

--rtmp **-r** *url*

URL of the server and media content.

--host **-n** *hostname*

Overrides the hostname in the RTMP URL.

--port **-c** *port*

Overrides the port number in the RTMP URL.

--protocol **-l** *number*

Overrides the protocol in the RTMP URL.

0 = rtmp

1 = rtmpt

2 = rtmpe

3 = rtmpte

4 = rtmps

5 = rtmps

--socks **-S** *host:port*
 Use the specified SOCKS4 proxy.

Connection Parameters

These options define the content of the RTMP Connect request packet. If correct values are not provided, the media server will reject the connection attempt.

--app **-a** *app*
 Name of application to connect to on the RTMP server. Overrides the app in the RTMP URL. Sometimes the rtmpdump URL parser cannot determine the app name automatically, so it must be given explicitly using this option.

--tcUrl **-t** *url*
 URL of the target stream. Defaults to rtmp[e]://host[:port]/app/playpath.

--pageUrl **-p** *url*
 URL of the web page in which the media was embedded. By default no value will be sent.

--swfUrl **-s** *url*
 URL of the SWF player for the media. By default no value will be sent.

--flashVer **-f** *version*
 Version of the Flash plugin used to run the SWF player. The default is "LNX 10,0,32,18".

--auth **-u** *string*
 An authentication string to be appended to the Connect message. Using this option will append a Boolean TRUE and then the specified string. This option is only used by some particular servers and is deprecated. The more general **--conn** option should be used instead.

--conn **-C** *type:data*
 Append arbitrary AMF data to the Connect message. The type must be B for Boolean, N for number, S for string, O for object, or Z for null. For Booleans the data must be either 0 or 1 for FALSE or TRUE, respectively. Likewise for Objects the data must be 0 or 1 to end or begin an object, respectively. Data items in subobjects may be named, by prefixing the type with 'N' and specifying the name before the value, e.g. NB:myFlag:1. This option may be used multiple times to construct arbitrary AMF sequences. E.g.
 -C B:1 -C S:authMe -C O:1 -C NN:code:1.23 -C NS:flag:ok -C O:0

Session Parameters

These options take effect after the Connect request has succeeded.

--playpath **-y** *path*

Overrides the playpath parsed from the RTMP URL. Sometimes the rtmpdump URL parser cannot determine the correct playpath automatically, so it must be given explicitly using this option.

--live **-v**

Specify that the media is a live stream. No resuming or seeking in live streams is possible.

--subscribe **-d** *stream*

Name of live stream to subscribe to. Defaults to *playpath*.

--start **-A** *num*

Start at *num* seconds into the stream. Not valid for live streams.

--stop **-B** *num*

Stop at *num* seconds into the stream.

--buffer **-b** *num*

Set buffer time to *num* milliseconds. The default is 20000.

--timeout **-m** *num*

Timeout the session after *num* seconds without receiving any data from the server. The default is 120.

Security Parameters

These options handle additional authentication requests from the server.

--token **-T** *key*

Key for SecureToken response, used if the server requires SecureToken authentication.

--jtv **-j** *JSON*

JSON token used by legacy Justin.tv servers. Invokes NetStream.Authenticate.UsherToken

--swfhash **-w** *hexstring*

SHA256 hash of the decompressed SWF file. This option may be needed if the server uses SWF Verification, but see the **--swfVfy** option below. The hash is 32 bytes, and must be given in hexadecimal. The **--swfsize** option must always be used with this option.

--swfsize **-x** *num*

Size of the decompressed SWF file. This option may be needed if the server uses SWF Verification, but see the **--swfVfy** option below. The **--swfhash** option must always be used with

this option.

--swfVfy **-W** *url*

URL of the SWF player for this media. This option replaces all three of the **--swfUrl**, **--swfhash**, and **--swfsize** options. When this option is used, the SWF player is retrieved from the specified URL and the hash and size are computed automatically. Also the information is cached in a *.swfinfo* file in the user's home directory, so that it doesn't need to be retrieved and recalculated every time `rtmpdump` is run. The *.swfinfo* file records the URL, the time it was fetched, the modification timestamp of the SWF file, its size, and its hash. By default, the cached info will be used for 30 days before re-checking.

--swfAge **-X** *days*

Specify how many days to use the cached SWF info before re-checking. Use 0 to always check the SWF URL. Note that if the check shows that the SWF file has the same modification timestamp as before, it will not be retrieved again.

Miscellaneous

--device **-D** *address*

Listener IP address. The default is 0.0.0.0, i.e., any IP address.

--sport **-g** *port*

Listener port. The default is 80.

--quiet **-q**

Suppress all command output.

--verbose **-V**

Verbose command output.

--debug **-z**

Debug level output. Extremely verbose, including hex dumps of all packet data.

--help **-h**

Print a summary of command options.

EXAMPLES

The HTTP request

```
GET /?r=rtmp:%2f%2fserver%2fmyapp&y=somefile HTTP/1.0
```

is equivalent to the **rtmpdump(1)** invocation

```
rtmpdump -r rtmp://server/myapp -y somefile
```

Note that only the shortform (single letter) options are supported.

ENVIRONMENT

HOME

The value of **\$HOME** is used as the location for the *.swfinfo* file.

FILES

\$HOME/.swfinfo

Cache of SWF Verification information

SEE ALSO

rtmpdump(1)

AUTHORS

Andrej Stepanchuk, Howard Chu, The Flvstreamer Team

<<http://rtmpdump.mplayerhq.hu>>