

This is the distribution kit for the stand-alone WriteLog internet server.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

\*\*\*\*\*

### **Building and/or Installing the Server:**

The server sources and a Linux and Win32 build of each are maintained on github:

<https://github.com/w5xd/WriteLogInetServer>

It has these directories:

- \bin
  - Contains ContestQsos.wsdl -- needed at run time for WriteLog version 10 clients.
- \bin\win32
  - Contains the win32 executable built with Visual Studio 2013
- \bin\linux32
  - build from source
- \source
  - Contains the C++ source files for the binaries. You may rebuild the server for platforms not included in this kit. cd to this directory, and type "make". The result is a WritelogInetServer executable in the source directory that you must manually copy to the bin/linux32 folder.

\*\*\*\*\*

### **Enhanced security at the server**

Connecting a program on your computer to the internet exposes your machine to increased chances of being compromised by outside hackers. Firewalls must be turned off, at least selectively for this program, and this program reads files off your computer and copies them to callers. You can vastly increase the security of the machine running this server by running the server in a virtual machine (VM). There are multiple vendors of VM software, but here is a free one: <http://www.virtualbox.org/>

The VM needs to host an operating system. Here is a free one you can install in a VM:

<http://www.ubuntu.com/>

If you use VirtualBox and Ubuntu, read the VirtualBox instructions on networking to see how to make the WriteLog internet server inside the VM visible outside the VM.

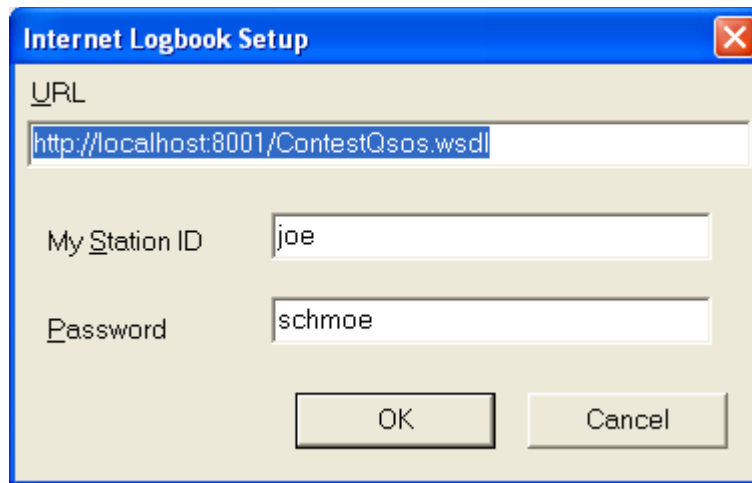
If you are running the server "from home" (from a DSL or cable subscription), your security almost requires using a router (which itself has NAT) to isolate your home network from the internet. That router will prevent the incoming connections necessary

for the WriteLog internet server, but most can be configured to allow designated ports through to designated machines. Read the router documentation. It is also possible that your ISP blocks incoming connections. In that case you will not be able to run a server.

\*\*\*\*\*

### From the client side

In WriteLog's "Tools Connect To Internet Logbook" menu dialog, there are 3 things you (the server administrator) must tell users to fill in. In that dialog, those three things are:



The screenshot shows a Windows-style dialog box titled "Internet Logbook Setup". It has a blue title bar with a close button (X) in the top right corner. The dialog contains three input fields. The first field is labeled "URL" and contains the text "http://localhost:8001/ContestQsos.wsd". The second field is labeled "My Station ID" and contains the text "joe". The third field is labeled "Password" and contains the text "schmoe". At the bottom of the dialog are two buttons: "OK" and "Cancel".

### URL

The url must start with <http://> The next thing in the IP address. It can be numeric, as in 192.168.100.1

or it can be a name, like:

myserver.mydomain.com

You have to know things about DNS to make this work. Note that services like this: <http://www.no-ip.com> can rent you a domain forwarding service, and they even have a free service with not many features, but enables publishing a logbook server from your home PC--if your home network AND your ISP are all configured to allow such publishing.

The domain name must be followed by a colon character and then a port number. The port number must be a number from 1 to 65535, but there are reasons to use various port numbers that are too detailed to document here. If you have no clue, try 8001, the default.

The port number must be followed by a slash character. For WriteLog version 10 clients, the slash character must be followed by the name of a .wsdl file that is located on your server. The one provided is named ContestQsos.wsd. Here is an example:

<http://192.168.1.100:8001/ContestQsos.wsd>

## My Station ID

The string the user entered here is used in two different places by the system:

- a) all QSOs logged have this string in the "INET" column.
- b) it becomes a "username" to be looked up in your server's password file (see below). If you have no password file (and none is provided in the installation kit), then any username and password is allowed.

## Password

If you have a password file at your server, the user must provide the password that matches the username (My Station ID), in your password file. If you have no password file, then the Password entry by users is ignored.

*Note that this server uses http basic authentication to send the username/password, which means they are not encrypted as transmitted over the internet. This will only keep "honest" people out of your server. Anyone with the ability to sniff your tcp packets between your server and your clients can read the usernames/passwords. Make up dedicated usernames and passwords for this application that you use nowhere else.*

\*\*\*\*\*

## Configuring the server.

It is designed to run with the process current directory set to the \bin installation directory. ContestQsos.wSDL (an edited and renamed version of it) must be in the process current directory.

### Editing the .wSDL file

This server does not use the .wSDL, but the WriteLog version 10 client requires it at run time, and it is served from WritelogInetServer. For WriteLog version 10 clients, ContestQsos.wSDL **must** be edited for your installation. At the bottom, you will find this line:

```
<soap:address location="http://192.168.1.101:8001"/>
```

You must change the address (the 192.168.1.101) to the IP address your clients are to connect to. You *may* (and usually *should*) use the TCP name rather than the numeric address if your server is located where DNS will have the right number. The number after the colon, :8001 in this example, must match the "-p 8001" used to launch this program. Note that WriteLog users must also enter the file name in their setup. That means the server *may* have multiple .wSDL files if you have the situation where some users route through, say, a NAT server and others are "inside" the NAT. Have two different WSDL files and tell each user which one to connect to.

### 4. A username/password file.

This file is optional. There is none provided; you have to create one with a text editor. If you create one, launch the executable with the -u parameter pointing to it. You may place it anywhere in the directory structure that you want, but be aware that if you put it in the \bin installation folder or any subfolder, the server will allow it to be downloaded by any internet browser. Not having a password file means no usernames or passwords are required to connect to the server.

More than one "user" is allowed to connect simultaneously using the same username and password combination, but, generally, you want to assign different ones to every connecting station because the username at the WriteLog end is logged with each QSO.

## 5. SHUTDOWN file

If there happens to be a file in the process current directory who's name contains the string SHUTDOWN, and if a user with a valid username/password uses a web browser to brose to that file, then this server will server out the contents of that file and then shut down. Note that files of any other name in this directory or any subdirectory are available to browsers.

\*\*\*\*\*

## Running the server

usage: WritelogInetServer [-h] [-u <userFile>] [-p <portNumber>] [-v]

-v

turns on the verbose flag that prints sent/received QSO counts.

-h

prints this help message

-u <userFile>

<userFile> is the path to a username/password file.

The file is formatted with one username/password pair per line.

Separate the username from the password with one space character.

These become the legal entries from WriteLog users in their

Connect to Internet Logbook dialog.

-p <portNumber>

This program requires a TCP port. The default without -p is 8001.

The number must be in the range of 1 through 65534. Different operating systems, internet providers, and routers will allow or not allow various port numbers through.

-t <number of threads>

number of threads to use to serve. Limited to 10 max. Default is 1.

-L6

turns on the limit-labels-to-6-characters flag, which then rejects invalid input

from WriteLog versions earlier than 10.76.

If WriteLog version 10.75 and earlier are participating in the network, then you should also add the "-L6" command option. This option causes the server to truncate network

frequency names longer than 6 characters, which those earlier WriteLog versions can send to the server.

That's all. If you add `-v` to the command line, it prints messages as it goes along. If it exits, you can simply restart it. The WriteLog clients will re-populate the logbook after it restarts. For unattended use, write a script that simply runs `WritelogInetServer` in a loop- if it crashes (which it should not), the script just starts it over. Such scripts are operating-system specific and not provided in this kit.

\*\*\*\*\*

### **Test the server from a browser**

A web browser can be used to test whether the server is running and can answer requests on its selected ports. Browse to the same URL that WriteLog users are instructed to connect to in WriteLog:

**`http:// myserver.mydomain.com:8001/ContestQsos.wsdl`**

The result should be a view of the `ContestQsos.wsdl` file as edited per these instructions. If you have configured the server to require passwords, your browser will ask for one of them before it will allow a download of the file.

\*\*\*\*\*

### **History**

May, 2016

- update CPP server to use synchronization and build using C++11 for both Linux and Win32.

WritelogInetServerKit06.zip                      December, 2011  
Support WriteLog version 11 clients.

WritelogInetServerKit05.zip                      May, 2011  
update documentation  
use timeouts to avoid hung server  
verbose option prints incoming IP address

WritelogInetServerKit04.zip                      February, 2010  
add -L6 option

WritelogInetServerKit03.zip                      February 2010  
fix 5 minute delay overwriting old network frequency with incoming.