WWF Analyzer

By: Kevin McNee

This program analyzes world writable files reports and summarizes the results.

**Table of Contents**

[1.0 Input Files 3](#_Toc330909808)

[2.0 Running for the First Time and Setting Preferences 4](#_Toc330909809)

[3.0 Performing the Analysis 5](#_Toc330909810)

[3.1 Running the Program 5](#_Toc330909811)

[3.2 Erroneous Input 8](#_Toc330909812)

[4.0 Output 9](#_Toc330909813)

[4.1 Details Report 9](#_Toc330909814)

[4.2 Summary Report 11](#_Toc330909815)

# 1.0 Input Files

The files that this program uses as input must be the same format as the results of an [ls command with the long format option](https://en.wikipedia.org/wiki/Ls). The program requires that each line in the file has one listed file and that it has the following data, separated by whitespace: UNIX file type and permissions (in [symbolic notation](https://en.wikipedia.org/wiki/Filesystem_permissions#Symbolic_notation)), number of hard links, owner, group, size, month, date, time/year, and filename. For example:

-rw-rw-rw- 1 unixguy staff 6074 Jan 01 2011 /home/unixguy/docs/example.txt

-rw-rw-rw- 1 unixguy staff 42 Jan 26 13:57 /home/unixguy/config.sh

Such a file can be generated with the following command:

find / -perm -o+w -type f -exec ls -l {} \; > MyServer-WWFiles-date.out

Note that this program infers the server’s name from the filename of the WWF report. The program considers every character before the first non-alphanumeric character to be part of the server name. For example, the WWF report “MyServer-WWFiles-01012011.out” is infered to be for server “MyServer”.

# 2.0 Running for the First Time and Setting Preferences

This program is run as a standalone executable file. It does not have to be run from any particular directory as long as the user is able to read from and write to this location.

When this program is run for the first time, the user will be presented with the notification that a preferences file (WWF Analyzer.pref) has been created in the same directory as the executable. The file is used to customize the output of the program. Open the file with a text editor (e.g. Notepad) and make any desired changes as described therein.

The following are changes that can be made in the preferences file (refer to the file for more details):

* Limit the number of servers displayed on the [summary report](#_4.2_Summary_Report)
* Limit the number of owners displayed on the [summary report](#_4.2_Summary_Report)
* Limit the number of high volume entries per owner or server on the [summary report](#_4.2_Summary_Report)
* Ignore certain files based on extension or substring
* Flag certain files as critical based on extension or substring
* Limit the number of critical files displayed on the [details reports](#_4.1_Details_Report)

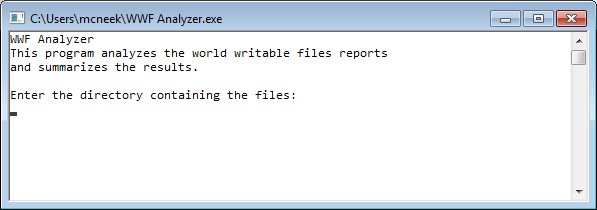
By default, none of the output is limited and no files are ignored or flagged as critical.

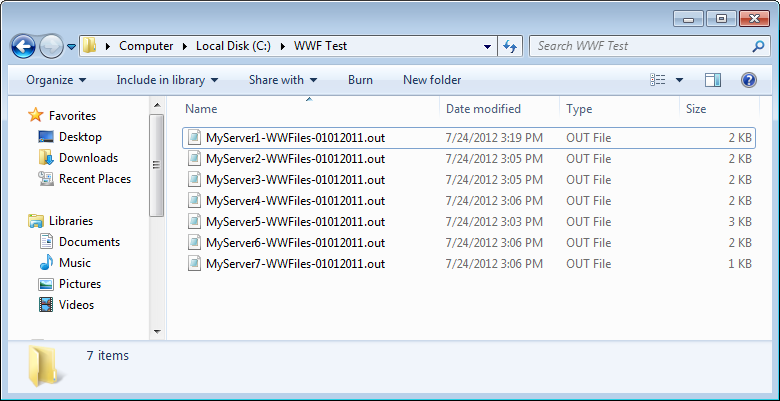
If a file is ignored, then it is neither counted towards the owner of the file, nor counted towards the server on which it is located. This feature is for files which are unimportant. If a file is flagged as critical, then it will be displayed in the details report of the server on which it is found and there will be an alert in the summary report. This feature is for identifying files which are potentially threatening. Note that ignoring a file takes precedence over flagging it as critical. For example, if we flag files with the substring “/home/” as critical and ignore “.log” files, then /home/unixguy/example.log” will be ignored.

# 3.0 Performing the Analysis

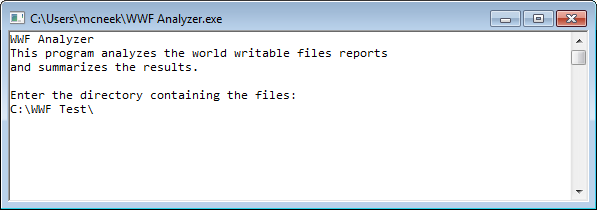
If the preferences file exists, the program will load the preferences and proceed to its normal operation.

## 3.1 Running the Program

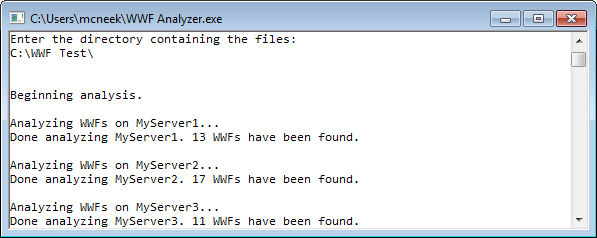
The user will be presented with the following screen: 

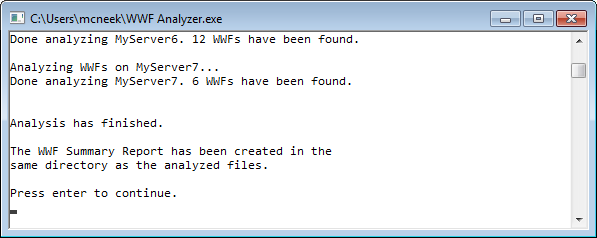
Locate the directory where the files are located. 

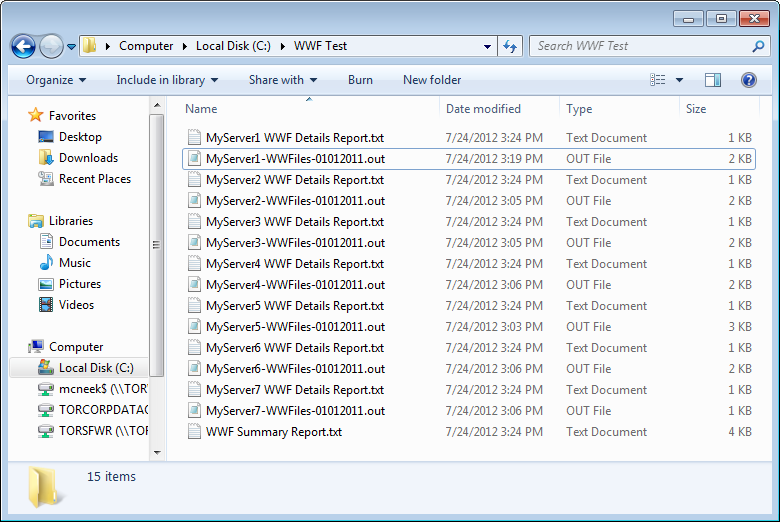
For this example, it is “C:\WWF Test\”.

Enter the path in the program. 

Conveniently, Windows paths are case-insensitive and accept forward slashes instead of backslashes. Also note that, for this program, the trailing slash is optional. So, “c:/wwf test” would also work.

When the path is entered, the program will begin its analysis using the files in the directory. 



Once the analysis has finished, the [details reports](#_4.1_Details_Report) and the [summary report](#_4.2_Summary_Report) will be found in the same folder as the files. 

When enter is pressed to end the program, the [summary report](#_4.2_Summary_Report) will be opened automatically.

## 3.2 Erroneous Input

If the directory that is entered does not exist, the program will prompt the user to enter a valid directory.

If the directory exists, but does not contain any files, the program will just report that no WWFs have been found.

If the directory contains files that do not match the criteria as described in [section 1.0](#_1.0_Input_Files_1), the program will most likely go through the files and find no WWFs, However, this behaviour is not guaranteed, so it is best to make sure that there are no extra files in the directory. There are, however, certain files that are ignored by the program:

* Hidden files
* Directories (the program does not recursively search subdirectories)
* Any of the [reports](#_4.0_Output) generated by this program

The program also does a quick sanity check to catch some invalid files. In this case, the user will be alerted and the file will not be analyzed.

The files should be in plain text and should not have any formatting applied to them. The formatting will most likely be caught by the sanity check. If not, correct interpretation of the file is not guaranteed, just like the files which do not match the criteria.

While this should not be a real concern in practice, there is a limit to the number of WWFs that this program can store. The datatype used to store the count of WWFs is an “unsigned long” which has a maximum value of at least 4,294,967,295. There are no guarantees that it can handle more. The input files would likely be hundreds of gigabytes. Although, if you somehow have more than 4,294,967,295 WWFs, you have bigger problems than the possibility of this program not working.

# 4.0 Output

This program produces two types of output: a details report for each server and a summary report. They are created in the same directory as [the one entered by the user](#_3.1_Running_the). The output in these reports can be customized in [preferences file](#_2.0_Running_for).

## 4.1 Details Report

The name of the server is shown, followed by the following counters:

Files Found: The total number of files in the input file.

Files Ignored: The number of files which are not included in the count of WWFs (set in the preferences file). These files are not included in the WWF counts for the following tables nor are they displayed if they are also flagged as critical.

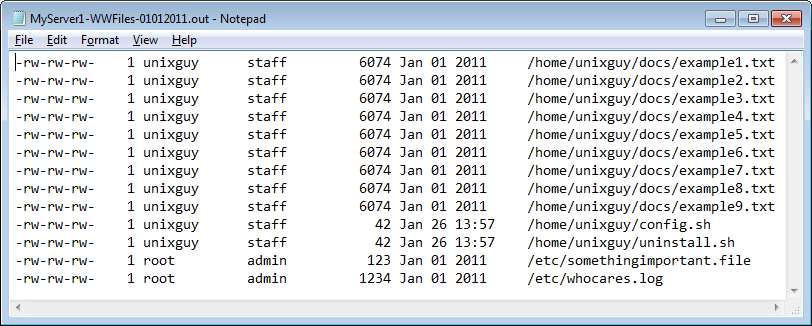
# of WWFs: The number of WWFs that are not ignored.

Critical Files: The subset of WWFs flagged as critical (set in the preferences file).

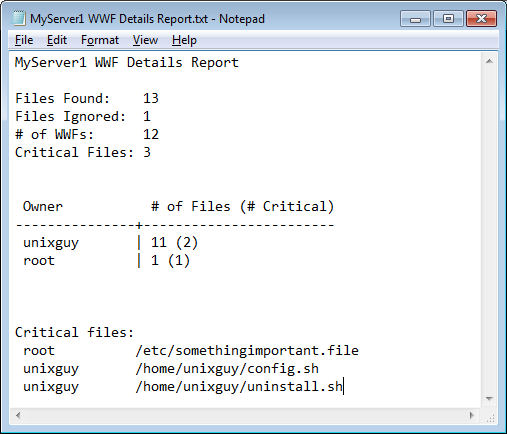
This is followed by a table with the owners of the WWFs and the number of WWFs that they own (sorted descending). The number of files that they own that have also been flagged as critical is displayed in parentheses. This is omitted if the number is 0.

If critical files have been found, they are listed. They are sorted by owner then by filename.

Here is an example:

We ignore “.log” files, and consider “.sh” and files which contain the substring “/etc/” to be critical. We use the following input file: 

This give the following result:



## 4.2 Summary Report

First, there is a table of the servers with the corresponding number of WWFs (sorted descending). It also shows the owners on this server with the most files. The same thing is done for owners. The number of WWFs, summed from all of the servers, for a given owner is shown as well as the servers with the most files.

If any critical files, on any of the servers, have been found, the server name is listed along with the number of critical files on that server (sorted descending by number of critical files then by server name). Refer to the identified server’s [details report](#_4.1_Details_Report) for more information.

Continuing from the [previous example](#_4.1_Details_Report) and adding in more servers, the following summary is produced:

