Command Line Test Tool

© 2019 Glasswall Solutions Ltd ALL RIGHTS RESERVED

Information contained herein is the property of Glasswall Limited and is proprietary and confidential.

Glasswall Solutions Ltd.

(e): support@glasswallsolutions.com

Creation Date – 21/06/13

Version – 2.11

**Document History**

Table 1: Document Change History

|  |  |  |
| --- | --- | --- |
| ***Issue Date*** | ***Issue Number*** | ***Description*** |
| 21/06/13 | 1.0 | Initial Draft |
| 03/07/2013 | 1.1 | Corrected item names in example config file to be consistent with naming through the document. |
| 05/09/2013 | 1.2 | Updated to describe the command line return values |
| 30/09/2013 | 1.3 | Added support for analyse, manage, and protect in single operation modes. |
| 02/10/2013 | 1.4 | Amended process mode value description to include the Analyse, Manage & Protect option 2.  Modified the command line description to strike through all options other than -config as they are not yet implemented |
| 15/10/2013 | 1.5 | INI for our Glasswall CLI tool requires a need to specify the file types that we are processing |
| 12/11/2013 | 1.6 | Updates to show file to file API calls now possible. |
| 11/03/2014 | 1.7 | Refinement to description for writeOutput flag. |
| 16/04/2014 | 1.8 | Updated the documentation to describe the -xmlConfigFile command line option and its associated file format. |
| 16/05/2014 | 1.9 | Updated to instruct that if no bitmap content management switches are specified in the ini file when not xml file is specified that GWFileConfig isn't called. |
| 27/08/2014 | 2.0 | Updated listed file type options to include Office XML file suffixes. |
| 03/09/2014 | 2.1 | Addition of logProcessStatus switch documentation |
| 08/09/2014 | 2.2 | CM bi-state configurations now deprecated. |
| 18/03/2015 | 2.3 | Conventions section added, and writeOutput and reportMode options clarified. |
| 22/04/2016 | 2.4 | Glasswall DLL Lite versions of manage protect. |
| 27/02/2017 | 2.5 | Updated the content management configuration XML example. |
| 29/09/2017 | 2.6 | Updated processMode to include 4 and 5. |
| 29/09/2017 | 2.7 | Added notes to processModes 4 & 5 regarding processMode. |
| 28/08/2018 | 2.8 | Extended xml config example to include recently introduced content management switches. |
| 18/11/2018 | 2.9 | Updated to include File type wildcard option and API mapping section. |
| 25/04/2019 | 2.10 | Removed File to File mode and added File Storage Mode |
| 13/11/2019 | 2.11 | Updated to reflect policy file (xmlconfig) is a mandatory parameter. |

# Contents

[1 Contents 4](#_Toc530562569)

[2 Conventions 5](#_Toc530562570)

[3 Usage 5](#_Toc530562571)

[3.1 Return values 5](#_Toc530562572)

[4 Configuration File Description 5](#_Toc530562573)

[4.1 Configuration Format 5](#_Toc530562574)

[4.2 Configuration Settings 6](#_Toc530562575)

[4.2.1 Section Name: GWConfig 6](#_Toc530562576)

[4.2.2 Mapping - CLI Modes To Glasswall API 8](#_Toc530562577)

[5 XML Configuration File Description 10](#_Toc530562578)

[5.1 Configuration Format 10](#_Toc530562579)

[5.2 Configuration Settings 11](#_Toc530562580)

[6 Test Tool Log File Format 12](#_Toc530562581)

[6.1 Files Log 12](#_Toc530562582)

[6.2 Application Log 12](#_Toc530562583)

[6.2.1 Header 12](#_Toc530562584)

[6.2.2 Configuration 12](#_Toc530562585)

[6.2.3 Body 12](#_Toc530562586)

[6.2.4 Footer 13](#_Toc530562587)

# Conventions

*Italic* – This is used to show arguments that must be replaced with user-supplied values.

**Bold** – This is used for commands and options that are to be typed in literally.

Constant width — used to show the contents of files or output from commands.

[ ] — these surround optional elements in a description of syntax. The brackets themselves should not be typed. Where brackets are to be typed literally, they will be in bold and shown in the examples.

# Usage

The Command Line Test Tool should be used in the following manner.

**commandlinetesttool.exe -config=***configFile* **-xmlconfig=***xmlConfigFile*

The contents of *configFile* and *xmlConfigFile* are described in the following sections.

## Return values

|  |  |  |
| --- | --- | --- |
| Name | Value | Description |
| rcSUCCESS | 0 | Test completed successfully |
| rcINVALIDCOMMANDLINE | 1 | Command line argument is invalid |
| rcDLLLOADFAILURE | 2 | Problem loading the DLL/Shared library |
| rcCONFIGLOADFAILURE | 3 | Problem loading the specified configuration file |
| rcPROCESSINGISSUE | 4 | Problem processing the specified files |

# Configuration File Description

This is passed to the **commandlinetesttool** executable using the **–config** option.

## Configuration Format

The configuration file is a text file that enables the operation of the test tool to be configured.

The file is made up of sections and associated configuration item name\value pairs.

For example

**[***sectionname1***]**

*itemname11* = *itemvalue11*

*itemname12* **=** *itemvalue12*

**[***sectionname2***]**

*itemname21* = *itemvalue21*

*itemname22* **=** *itemvalue22*

Item name must be unique within a section. Zero or more whitespace characters are allowed around the literal **=**.

An example valid configuration file is shown below

[GWConfig]

processMode=0

reportMode= 1

fileStorageMode=0

fileType = pdf

inputLocation= C:\Test\bug\_301

useSubfolders=1

outputLocation= Report

createOutputFolders=1

nonConformingDirName= NonConforming

managedDirName= Managed

quarantineNonconforming= 1

writeOutput= 1

logFileSize=0

logFileProcessTime=1

logProcessStatus=0

## Configuration Settings

### Section Name: GWConfig

All configuration value default to either ‘0’ or an empty string, where appropriate.

|  |  |  |
| --- | --- | --- |
| Identifier | Value | Notes |
| fileType | [pdf|jpg|png|gif|doc  |ppt|xls|docx|pptx|xlsx|emf|wmf|\*] | Type of documents for processing. Specifying the wildcard option ‘\*’ will enable automatic file type detection. Process mode 5 does not support the wildcard option. |
| inputLocation | [valid path] | Location of documents for processing |
| useSubfolders | [0|1] | 0 = Process documents only in inputLocation  1 = Process documents in inputLocation and any subfolders |
| processMode | [0|1|2|3|4|5] | 0 = Analysis  GWFileAnalysisAudit  GWFileAnalysisAuditAndReport  GWFileToFileAnalysisAudit  GWFileToFileAnalysisAuditAndReport  1 = Manage & Protect  GWFileProtect  GWFileProtectAndReport  GWFileToFileProtect  GWFileToFileProtectAndReport  2 = Manage & Protect Lite  GWFileProtectLite  GWFileProtectLiteAndReport  GWFileToFileProtectLite  GWFileToFileProtectLiteAndReport  3 = Manage & Protect & Report  GWFileAnalyseProtectReport  GWFileToFileAnalyseProtectReport  4 = Export Mode  Export images and text to an Interchange Package in the form of a ZIP archive. reportMode must be set to 0.  5 = Import Mode  Import images from an Interchange Package. The expected format is a ZIP archive. reportMode must be set to 0.  Note: The Short form of Report produced has no content items listed; only Sanitisations/Issue and Remedies |
| reportMode | [0|1] | 1 = Invokes the APIs that also produce a report. For example, GWFileAnalysisAuditAndReport. If **writeOutput** is set to zero, no reports will be produced. |
| writeOutput | [0|1] | 0 = No managed document or analysis report is produced  1 = Managed document and/or analysis reports are produced where appropriate and if **reportMode** is 1. |
| quarantineNonconforming | [0|1] | 1 = copy non-conforming document to quarantine folder. |
| outputLocation | [valid path] | Root folder for all processing output. |
| createOutputFolders | [0|1] | Creates output folders if they don’t exist |
| nonconformingDirName | [valid folder name] | Name of subfolder within **outputLocation** that is the destination for all output from the processing of non-conforming files. |
| managedDirName | [valid folder name] | Name of subfolder within **outputLocation** that is the destination for all output from the processing of managed files. |
| fileStorageMode | [0|1|2] | 0 = File to File  1 = File to Memory  2 = Memory to Memory |
| logFileSize | [0|1] | 1 = Includes the file size in the process log |
| logFileProcessTime | [0|1] | 1 = Includes the file processing time in the process log |
| logProcessStatus | [0|1] | 1 = Includes the process status information (from GWFileProcessMsg and GWFileProcessStatus) in the process log. This is an optional flag that defaults to 0 to denote no process status logging. |

### Mapping - CLI Modes To Glasswall API

The CLI tool can be used to verify the output of API functions in the Glasswall library. This section contains information on which CLI settings trigger specific API function calls.

The table below displays the API functions that are called under the various CLI configuration options.

|  |  |  |  |
| --- | --- | --- | --- |
| API Name | Process Mode | Report Mode | File Storage Mode |
| GWFileAnalysisAudit | 0 | 0 | 0 |
| GWFileAnalysisAuditAndReport | 0 | 1 | 0 |
| GWFileToFileAnalysisAudit | 0 | 0 | 1 |
| GWFileToFileAnalysisAuditAndReport | 0 | 1 | 1 |
| GWFileProtect | 1 | 0 | 0 |
| GWFileProtectAndReport | 1 | 1 | 0 |
| GWFileToFileProtect | 1 | 0 | 1 |
| GWFileToFileProtectAndReport | 1 | 1 | 1 |
| GWFileProtectLite | 2 | 0 | 0 |
| GWFileProtectLiteAndReport | 2 | 1 | 0 |
| GWFileToFileProtectLite | 2 | 0 | 1 |
| GWFileToFileProtectLiteAndReport | 2 | 1 | 1 |
| GWFileAnalysisAndProtectAndReport | 3 | 1 | 0 |
| GWFileToFileAnalysisProtectAndReport | 3 | 1 | 1 |
| GWFileToFileAnalysisProtectAndExport | 4 | 0 | 1 |
| GWFileToMemoryAnalysisProtectAndExport | 4 | 0 | 0 |
| GWFileToFileProtectAndImport | 5 | 0 | 1 |
| GWFileToMemoryProtectAndImport | 5 | 0 | 0 |
| GWMemoryToMemoryProtect | 1 | 0 | 2 |
| GWMemoryToMemoryAnalysisAudit | 0 | 0 | 2 |

Setting the FileType option to ‘\*’ will enable Glasswalls file type detection capability via a call to the **GWDetermineFileTypeFromFile** APIfunction (See the ‘Configuration Settings’ section). This will enable CLI to automatically determine the file type for each file that is about to be processed.

Supplying the ‘–h’ option to the CLI tool will display a help menu that provides additional help information as well as instructions on how to run the tool to produce the output for the following additional API functions.

* **GWGetAllIdInfo** – This retuns an XML file containing the descriptions for all possible issue IDs in the Glasswall engine.

**Command:**

*>> commandlinetesttool.exe -a output\_xml.xml*

* **GWGetIdInfo** – This returns a group description for a given issue ID.

**Command**:

*>> commandlinetesttool.exe -b 96 outputstring.txt*

*Where 96 is the issue ID and outputstring.txt is the file in which to store the output description.*

* **GWFileVersion –** Displays the library version number

**Command:**

*>> commandlinetesttool.exe -v*

The following API functions are called by the CLI tool when generating the Glasswall CLI process log.

* **GWFileErrorMessage** – The output of this is displayed along side each processed file if a file is marked non-compliant within the process log generated everytime a set of files are processed.
* **GWFileProcessStatus** – The output of this is displayed along side each processed file in the process log if the ‘logProcessStatus’ option is set to 1 (See the ‘Configuration Settings’ section).
* **GWFileProcessMessage** – The output of this is displayed along side each processed file in the process log if the ‘logProcessStatus’ option is set to 1 (See the ‘Configuration Settings’ section).

For more information on each API function please refer to the Glasswall SDK documentation.

# XML Configuration File Description

## Configuration Format

The format configuration within this file is described formally in the XSD located in the glasswall sdk documentation. This configuration is passed unchanged to the GWFileConfigXML function. **An example of a full configuration file is shown below**

<?xml version="1.0" encoding="UTF-8"?>

<config>

<pdfConfig>

<watermark>Glasswall Protected</watermark>

<acroform>sanitise</acroform>

<metadata>sanitise</metadata>

<javascript>sanitise</javascript>

<actions\_all>sanitise</actions\_all>

<embedded\_files>sanitise</embedded\_files>

<embedded\_images>sanitise</embedded\_images>

<internal\_hyperlinks>sanitise</internal\_hyperlinks>

<external\_hyperlinks>sanitise</external\_hyperlinks>

</pdfConfig>

<wordConfig>

<macros>sanitise</macros>

<metadata>sanitise</metadata>

<embedded\_files>sanitise</embedded\_files>

<embedded\_images>sanitise</embedded\_images>

<review\_comments>sanitise</review\_comments>

<internal\_hyperlinks>sanitise</internal\_hyperlinks>

<external\_hyperlinks>sanitise</external\_hyperlinks>

<dynamic\_data\_exchange>sanitise</dynamic\_data\_exchange>

</wordConfig>

<pptConfig>

<macros>sanitise</macros>

<metadata>sanitise</metadata>

<embedded\_files>sanitise</embedded\_files>

<embedded\_images>sanitise</embedded\_images>

<review\_comments>sanitise</review\_comments>

<internal\_hyperlinks>sanitise</internal\_hyperlinks>

<external\_hyperlinks>sanitise</external\_hyperlinks>

</pptConfig>

<xlsConfig>

<macros>sanitise</macros>

<metadata>sanitise</metadata>

<embedded\_files>sanitise</embedded\_files>

<embedded\_images>sanitise</embedded\_images>

<review\_comments>sanitise</review\_comments>

<internal\_hyperlinks>sanitise</internal\_hyperlinks>

<external\_hyperlinks>sanitise</external\_hyperlinks>

<dynamic\_data\_exchange>sanitise</dynamic\_data\_exchange>

</xlsConfig>

</config>

**Note:** ThexlsConfig, pptConfig, and wordConfig cover both office XML and office binary file types.

## Configuration Settings

|  |  |  |
| --- | --- | --- |
| Type | Value | Notes |
| contentManagementFlag | sanitise | Configures Glasswall to remove or clean document element types associated with this content management switch type from any document being processed. This removal will be logged in analysis reports as a ‘sanitisation item’ |
| allow | Configures Glasswall to leave document element types associated with this content management switch type in any document being processed. |
| disallow | Configures Glasswall to raise an issue if document element types associated with this content management flag are found within any document being processed. |
| watermark | Freeform text, up to 20 characters | Specifying a null element .i.e., <watermark />, results in no watermark being added to documents being processed. |

# Test Tool Log File Format

## Files Log

Produced by the test application before the processing of the files is started. Compiled from the specified input location, documenting the complete list of files to be processed.

The format of each line follows the same specification.

**==***nnnnn***==** *Filename*

One file is listed on each line, where *nnnnn* is a zero indexed counter that increments by one for each line.

## Application Log

Produced by the test application recording the results of processing. The file structure is made up of a header, body and footer.

### Header

* Line of **\*** characters
* Time date stamp for start of logging
* Description of library version in use
* Line of **\*** characters

Example

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

START LOGGING 14:24:08(15/04/2013)

SOFTWARE VERSION PLUGIN\_GLASSWALL\_DLL:ANALYSE\_AUDIT.MANAGE.PROTECT.01.01..BUILD(Apr 15 2013 09:43:25)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

### Configuration

A record of the configuration settings being used by the Glasswall DLL during testing. The information is provided in the XML format used when configuring the test. The configuration is provided by the DLL from the GWFileConfigGet function, rather than being a copy of what has been passed into the test tool.

### Body

Line per file, each line consisting of

* An index number, matching the index number used in the ‘Files Log’ described above.
* File name (preceded by a space)
* Test Result (preceded by a space and a horizontal tab)
* Processing Duration (preceded by two horizontal tabs)
* Optional file size (preceded by two horizontal tabs) (*see Configuration)*
* Optional Process Status (preceded by two horizontal tabs) and Process Status Message (preceded by two horizontal tabs) (*see Configuration)*

Example

==00000== Found file Golden\_90ms-RKSJ-H\_10.pdf Passed (460 ms) (2500 MB)

Processing Status 96 Processing Message "Remedies Required, Sanitisation Required"

### Footer

* Line of **\*** characters
* Time date stamp for end of logging
* Total duration of processing (calculated as delta between the start and end logging times)
* Line of **\*** characters

Example

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

END LOGGING 15:24:08(15/04/2013)

DURATION 01:00:00

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*