Monitor file integrity using MultiHasher

Introduction

This guide describes the use of MultiHasher, an integrity monitoring tool for Microsoft Windows that may be used to monitor data files for evidence of accidental or deliberate change. Unexpected file change may be a symptom of an underlying problem – that the storage media is beginning to fail, or that the system has been compromised and updates made by a non-authorised user. For this reason it is important to monitor the integrity of valuable files on a regular basis.

What is File Integrity Monitoring?

File Integrity Monitoring (FIM) is a method commonly applied in the I.T. sector to monitor data files held on digital media, identifying which files remains the same and which have changed.

A FIM tool works by apply a hash algorithm to one or more data files and recording the output. The output, referred to as a 'hash sum' or 'checksum' acts as a digital finger print that may be used to identify the file. For example, '`a4ce993d1974374d13ae48926eb478a57734f780`' represents the MD5 hash sum of a specific PDF document. By producing a hash sum of data files at regular intervals and comparing it to one produced at an earlier date, a user can monitor the file's integrity. If the file is unchanged, the hash sum will remain the same. If, however, it is modified, a different hash sum value will be produced.

How does File Integrity Monitoring work in practice?

The Integrity Monitoring workflow is comprised of a number of activities and decisions. These are illustrated in Figure 1.

Library & Archives Service
www.lshtm.ac.uk/library/
library@lshtm.ac.uk
+44 (0)20 7927 2276

LONDON SCHOOL of HYGIENE &TROPICAL MEDICINE

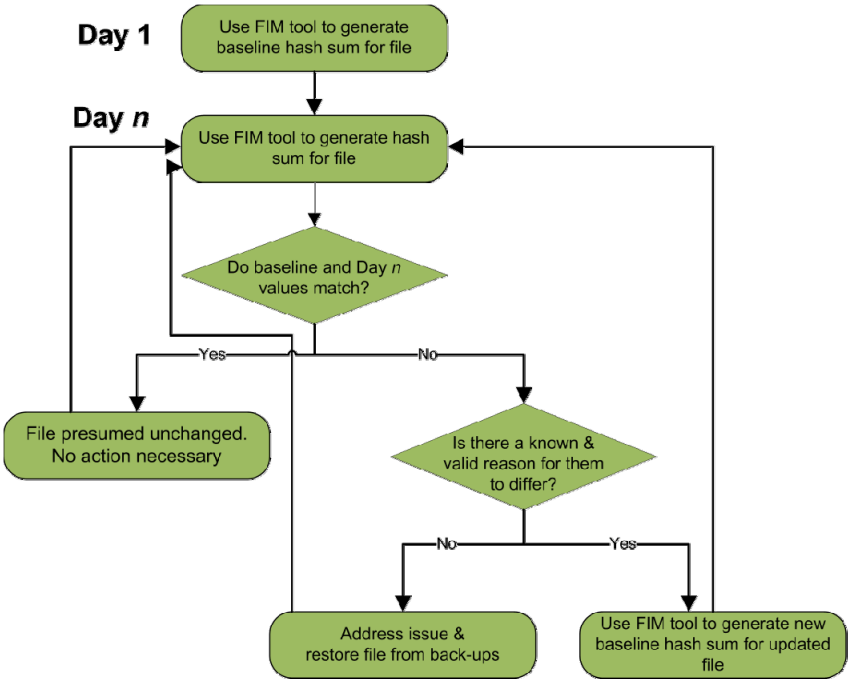

Figure 1: Integrity monitoring process

File Integrity Monitoring should be performed at regular intervals. Best practice in this area dictates that integrity monitoring should be performed on institutional systems every 24 hours – a practice that LSHTM's IT Services apply to all central storage that they maintain. For researchers' wishing to manage their own storage devices, such as portable hard disks and laptops, it is advisable to perform integrity monitoring activities at least once every month.

Further information on File Integrity Monitoring can be found on the Research Data management website at http://www.lshtm.ac.uk/research/researchdataman/.

What is MultiHasher?

MultiHasher is a free File Integrity Monitoring tool written Microsoft Windows. It may be used to calculate hash sums for one or more data files held on a disk and determine which, if any, have been modified since the previous check. The tool may be configured and executed using a graphical interface or via the command line. It supports a number of algorithms, including CRC32, MD5, SHA-1, SHA-256, SHA-384, and SHA-512.

Further information on the tool, including download links, is available from http://www.abelhadigital.com/multihasher

1. Generate hash sums for a set of files

This tutorial describes the steps to produce a hash sum for a set of files held in a folder on some form of storage media. The hash sum list will be used as a baseline to determine which files, if any, have been subsequently modified. Instructions on verifying these hash sums are outlined in Stage 2.

MultiHasher will display a configuration window, similar to Figure 1, on launch. This window may be used to configure various options and initiate activities to be performed.

1. The first step is to set the algorithms that will be used to document and verify each data file. Press the 'MD5, SHA-256, SHA-512' button to display a list of available hash algorithms.

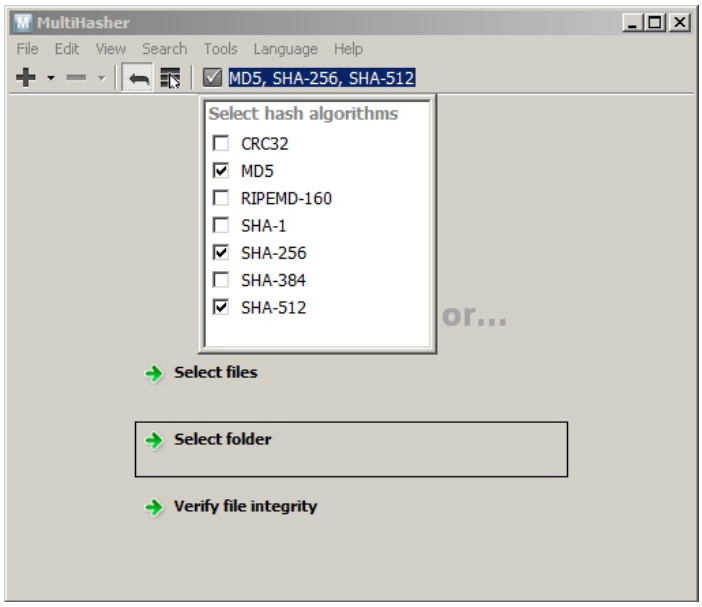

Figure 2: Hash algorithms available in MultiHasher

2. Select one or more hash algorithms to be used by clicking the box to the left of each option.
    a. For increased security, it is advisable to select two or more algorithms
    b. A strong hash algorithm, such as SHA-256, 384, or 512, should be used. These take more time to generate than CRC32 and MD5, but are considered to be more secure.

3. Press the 'Select folder' button to specify the directory to be analysed. Press OK.

    A processing screen will be displayed, indicating the number of files processed and remaining. File integrity calculations may be cancelled at any time by pressing the STOP button.

4. Once complete, the hash sum list (.mhx file) should be saved to disk.

    a. Click the 'File' menu and navigate to the 'Save list' menu item.

    b. Select a directory in which you wish to save the hash sum list.
        - It is advisable to store the .mhx file with the files that have been checked, e.g. in the same directory or a sub-directory. Multihasher uses relative paths, identifying file location in relation to the location of

the .mhx file. If you store the file in a different location or move it at a later date, Multihasher may not be able to find files to be checked.

c. Specify the filename to be used
   - It is advisable to record the collection name and date that you performed the baseline analysis in the filename. For example, 'msc_project_hashsum_*2014-01-01.mhx*'.

5. The Multihasher Hash list (.mhx) file is stored as XML using a custom scheme, as shown in Figure 3. It may be viewed using any text viewer.

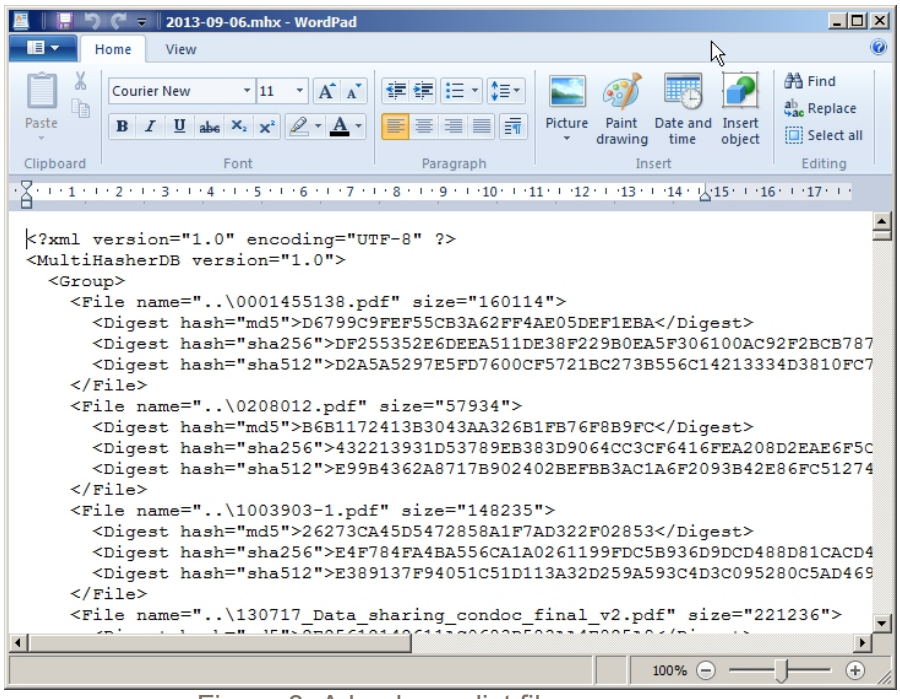

Figure 3: A hashsum list file

2. Verify hash sums to establish whether files are the same or different

The verification process is performed to determine which files, if any, have changed since the baseline integrity check was performed.

For personal backups, it is advisable to perform integrity monitoring activities at least once every month. The process may be automated using a time-based scheduling tool, such as the Windows Scheduler, Cron tool and others.

1. Press the 'Verify file Integrity' button and select a .mhx file to use as a baseline.
   - If the files to be analysed are found, a file count will appear in the bottom-left of the window indicating the number of files to be processed.

2. Review the output of the integrity comparison, once it has been completed (as shown in Figure 4)
   - The 'All' menu item displays a list of processed files.
   - The 'Invalids' menu item displays a list of files that have failed hash sum comparison.
   - The 'Error' menu item displays a list of files that MultiHasher has been unable to analyse

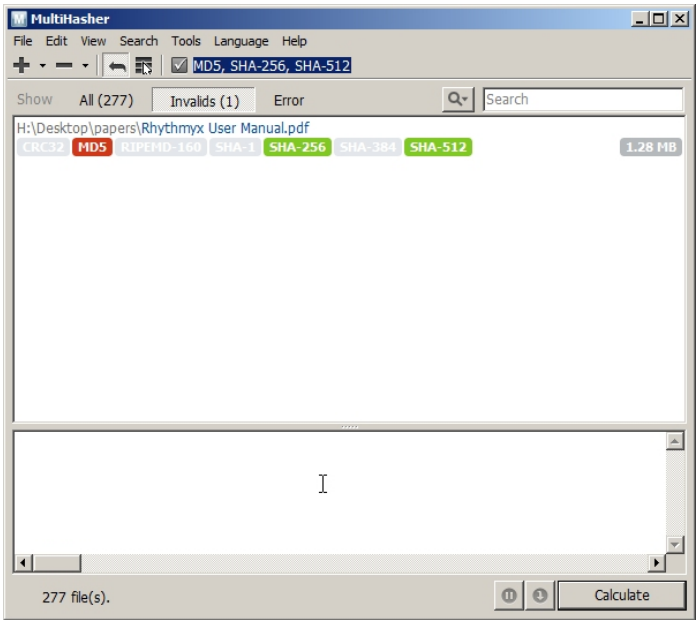

Figure 4: Verification process

'Invalid' hash sums and error messages should be investigated and the cause determined:

- An Invalid hash sum report that displays one or more red labels and no green labels may indicate the file has been modified. The user should determine if the update was performed for a legitimate purpose.
- An Invalid hash sum report that displays multiple hash sum algorithms, of which only one is red (similar to Figure 5) may indicate that the .mhx file itself has been modified.
- An error report stating that one or more files cannot be checked may be caused for several reasons. These include: the application does not have permission to read the file or the storage media is beginning to fail.

In many cases, files are updated for legitimate reasons and changes to the hash sum can be ignored. However, unexpected file change may be a symptom of an underlying problem – that the storage media is beginning to fail, or that the system has been compromised and updates made by a non-authorised user. For this reason it is important to monitor the integrity of valuable files on a regular basis.

How do I get more help?

The LSHTM Research Data Management Support Service provides advice and guidance on topics related to the creation, management, and sharing of research data. Information material and contact details are available at http://www.lshtm.ac.uk/research/researchdataman/.