Data Formats
HashSets.com Databases

Updated: 20 December 2021

1. Introduction

This document describes the formats of various database tables provided by HashSets.com to customers who subscribe to our ‘Database Exports’ (also known as ‘database dumps’). Throughout this document we will cover only trusted, known-good and/or non-threatening hash set tables from our database.

Since November 2003, Whitehat Computer Forensics, LLC, has been performing hash file calculations of common electronic files found within various types of computer operating systems, workstations and servers.

Our largest hash set consist of operating system files from more than 500+ operating system versions we have installed and forensically analyzed since 2003. The process involves installing the operating systems onto sterile hard drives and/or virtual disks, performing analysis using commercial computer forensic software to discover all files, file attributes and meta-data and most importantly the MD5, SHA1 and SHA-256 hash values.

All of the aforementioned findings are then imported into individual operating system tables which subsequently form our HashSets.com database and online search engine. We concurrently offer to our Gold and Platinum subscription members the database tables exported into tab delimited text files which may be reimported into their own database or third-party software products. Each table also provides a header row for simple field name explanation and identification (see further below for details).

2. Database Overview of the ‘Primary Database Export’ Table

The HashSets Database consists of thirteen very large database tables. From these tables there is one key table named ‘**Primary Database Export**’ table.

The ‘**Primary Database Export**’ table contains information that does not change for every file we have gathered and analyzed including a file’s corresponding MD5, SHA1, SHA-256 hash values, the same file’s initial 32 bytes in hexadecimal, the first 128 ASCII characters (similar to ‘strings’), logical byte file size, and so on.

The ‘**Primary Database Export**’ table contains hash values of only non-threatening, known-good and computer safe files. There are also NO duplicate hash values in this table. This is considered the KEY table that would be associated with the remaining twelve (12) tables mentioned discussed further down below.

‘Primary Database Export’ Table Schema

Field Name	Type	Length	Description
MD5	char	32	128-bit Message Digest 5 (hash value) of a specific file.
SHA_1	char	40	160-bit Secure Hash Algorithm message digest (hash value) of a specific file.
SHA_256	char	64	256-bit Secure Hash Algorithm message digest (hash value) of a specific file.
Header_HEX	varchar	64	First 32 bytes in Hexadecimal format of the file.
128_Bytes_ASCII	varchar	128	First detectable 128 bytes in ASCII format (Similar to performing STRINGS).
Signature	varchar	255	Suspected or possible file signature (header information).
Logical_Size	bigint	17	File size in byte format.
NSRL	char	3	File known to be found within the National Software Reference Library (NSRL) Dataset released by the US Government.
key_field	int	11	(Primary Database Key, if provided.)
SHA_512	char	128	512-bit Secure Hash Algorithm message digest (hash value) of a specific file, if known.
CRC32	char	8	32-bit Checksum of a specific file, if known.
Fuzzy_SSdeep	Varchar	255	(Future Use Only/Work in Progress. Context Triggered Piecewise Hash values (CTPH). Also called “fuzzy” hash values)
Ignorable	char	7	Intended to identify ignorable hash values such as files that are unique to one particular instance only (e.g. operating system log files, etc). Available choices are ‘Yes’, ‘No’ and ‘Unknown’)

3. All Other Remaining Tables Described

The remaining twelve (12) tables are divided into operating systems (MS Windows, Linux, BSD, macOS and Solaris) and non-operating system groups (MS Windows App Store, Mac App Store, etc).

Below is a description of each of the twelve (12) remaining tables. It should be mentioned briefly that over the course of many years (18+ years) the size of MS Windows and macOS operating system tables grew too large to contain within one table each. Therefore, in July 2021, we decided to divide those tables into smaller tables for convenience and ease of importation. The MS Windows tables were ultimately divided into three smaller tables (Windows North America, Windows Europe and Windows

Asia). For macOS we divided the previous single table into two tables (macOS 8 thru 10 and macOS 11 and above).

- **Windows_North_America_Database_Export** – This database table contains various Microsoft Windows operating system versions for the US and Canada (French Canadian) and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **Windows_Europe_Database_Export** – This database table contains various Microsoft Windows operating system versions for Europe (West and East) and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **Windows_Asia_Database_Export** – This database table contains various Microsoft Windows operating system versions for Asia (Japan, China and so on) and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **Linux_Database_Export** – This database table contains various Linux operating system distributions and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **MacOS_8_thru_10_Database_Export** – This database table contains various Apple macOS 8 and 9 (legacy) and macOS 10 (OS X) operating system versions and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **MacOS_11_and_above_Database_Export** – This database table contains Apple macOS 11 and above operating system versions and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **BSD_Database_Export** – This database table contains various BSD (UNIX like) operating system distributions and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **Solaris_Database_Export** – This database table contains various Solaris operating system versions and the individual file or folder names, file name extensions, last Modified/Accessed/Created file dates, and so on. Basically, the individual file or folder information that may not remain the same throughout the purpose of a MS Windows operating system regardless of individual file hash values (MD5, SHA1 and SHA-256) that generally remain the same until the file itself is ultimately changed or modified.

- **Applications_and_Hardware_Software_Drivers_Database_Export** – This database table contains the details of file gathered from third-party utilities and software applications as well as software drivers from common hardware manufacturer websites. Due to the large number of files within this category the installation process before hashing is not be performed. As a substitute the analysis of individual files incorporated a file unpacking process, when possible, then the gathering of hash value calculations.

- **Windows_App_Store_Database_Export** – This database table contains MS Windows 8 and 10 Applications commonly found within the MS Windows App Store. Specifically, downloadable business, game, education and other apps which were installed, analyzed and then gathered into MD5, SHA-1 and SHA-256 hash sets.

- **Mac_App_Store_Database_Export** – This database table contains macOS applications commonly found within the macOS App Store. Specifically, downloadable business, game, education and other apps which were installed, analyzed, hashed and then gathered into MD5, SHA-1 and SHA-256 hash sets.

- **US_Government_Database_Export** – This optional database table contains common non-threatening known hash values consisting of US Government (federal, state, local and military) publicly accessible website images, logos, multimedia files, office documents (.doc, .pdf, .xls, .ppt, etc). Please note that this table may be removed completely sometime in the future as these formerly common hash values may no longer be of use or popular to computer forensic or computer security professionals. For the time being we have furnished this table for optional download and use.

Other Optional Table(s):

- **File_Extensions** – This table contains general information or descriptions pertaining various file name extensions (.exe, .pdf, .dll, etc).

  For example, the below file extension .EXE within this table could be described as one of the following from historical popular use:

  EXECUTABLE FILE :::: SELF-DISPLAYING IMAGE :::: SELF-EXTRACTING ARCHIVE :::: SETTLERS 4 SAVE FILE :::: PDP-10 PAGE-MAPPED EXECUTABLE BINARY FILE :::: PLAYSTATION EXECUTABLE FILE :::: OUT-OF-PROCESS CODE COMPONENT FILE :::: MICROSOFT LINKER EXE INPUT FILE EXTENSION :::: DATAFLEX RUNTIME FILE

EXTENSION :::: SELF-EXTRACTING ARCHIVE :::: MIME: APPLICATION/OCTET-STREAM FILE EXTENSION :::: MIME: APPLICATION/X-MSDOWNLOAD

This information above and within the table is used only as one of many starting points when analyzing computer files. It is not to be used as a guarantee that a particular file with a specific file extension is truly associated with any particular software, program, third-party utility, hardware device, etc.

4. Operating Systems and Non-operating System Table Schema

The following depicts the data elements for the previously mentioned twelve (12) tables. To associate the ‘Primary Database Export’ table with any of the above twelve (12) tables you would use the MD5 hash value from both tables as the linking “Key”.

Field Name	Type	Length	Description
MD5	char	32	128-bit Message Digest 5 (hash value) of a specific file.
Name	varchar	255	Name of a specific file that had been collected, analyzed and hashed.
File_Ext	varchar	255	The file name extension, if applicable.
Description	varchar	75	General file description only.
Last_Accessed	datetime	0	Date and Time that the file was last accessed at some point in time.
File_Created	datetime	0	Date and Time that the file was created at some point in time.
Last_Written	datetime	0	Date and Time that the file was last modified at some point in time.
Full_Path	text	0	A file’s Path or Directory as discovered during initial discovery analysis.
Quick_Category	varchar	75	Internal Use Only: An adhoc term used to catalog a file into some form of initial grouping.
File_Notes	varchar	255	Internal Use Only: Local analysis notes only mentioned during the initial collection and analysis of the file.
Major	varchar	75	Internal Use Only: Highest grouping description of a file.

Minor	varchar	75	Internal Use Only: Subgrouping description of a file.
Operating_System	varchar	75	Name of the associated operating system, if applicable.
Manufacturer	varchar	150	Name of the manufacturer, if known or applicable.
Version	varchar	50	Version name, if known or applicable.
Inside_Compressed_Files	char	7	(This specific description is to be removed in the future)
Record_Date	date	0	UTC time and date record was last modified (MS Windows only).
Is_Deleted	char	7	This particular file was found to be deleted or not deleted on the analyzed file system, if applicable.
key_field	bigint	15	(Primary database table key, if provided. Most databases require a Primary Key)
website	varchar	75	Website source, if applicable.
Geographic_Location	varchar	50	Global source location of the file's manufacturer, if applicable.
Extraneous	char	3	(Future Use /Work in Progress: Intended to identify ignorable files that are unique to one particular situation or instance e.g. system log files, registry files, etc. Available choices are 'Yes' or 'No')
Log	char	3	(Future Use Only/Work in Progress: Intended to identify suspected log files)
Graphic	varchar	25	(Internal Use Only: HashSets.com graphic symbol name to be associated with an icon or image for our HashSets.com Search Engine)
Bad_Extension	varchar	255	File found to have a suspected bad file extension in comparison to the file's signature/header information.
actual_file	varchar	255	Actual file and not some other type of similar like file such as those found within compressed files specifically, etc.

file_class	varchar	255	Class of file such as Regular File, Symbolic Link, etc.
folder	varchar	255	Identified as a file folder.
category	varchar	255	Specific type of file.
From_Recycle_Bin	varchar	255	Found within MS Windows Recycle Bin, if applicable.
From_Free_Space	varchar	255	Found within the free space on a drive.
moved_from_location	text	0	(Future Use Only)
recycle_bin_original_name	text	0	Original name of file from within Recycle Bin.
compressed	varchar	255	Found as a compressed like file, if applicable.
compressed_file_size	bigint	15	Size in bytes if found as a compressed file.
Compression_Method	varchar	255	Examples include Deflated, Stored, etc.
Extract_Version	varchar	255	Extraction version, if identifiable.
permissions	varchar	255	Primarily UNIX like read/write/execute permissions, if identifiable.
UID	int	255	User ID, if identifiable.
Group_Name_UNIX	varchar	255	UNIX like Group Name, if identifiable.
GID	int	255	Group ID, if identifiable.
Username	varchar	255	Username, if identifiable.
Container	varchar	255	Examples include Zip, GZIP, etc.
Encrypted	varchar	255	Found to have Encryption file like qualities.
Deleted_Date	datetime	0	Date file was known to be deleted as reported by the operating system, if applicable.
inode_number	bigint	15	Inode Number associated with *NIX like systems, if identified.
checksum	varchar	255	Checksum or hash sum computed on compressed files.
Hash_Search_Engine_Record_Date	date	0	(Internal Use Only: Date the file information was subsequently added into our search engine database by HashSets.com and/or WhiteHat Computer Forensics, LLC.)

5. In Summary

If, for whatever reason, you run into any issues or problems understanding the aforementioned database table structures, data or field types then please feel free to reach out to us via our website HashSets.com. We will make every effort to provide you with a reasonable amount of additional information to help you better understand the database tables.