



GSATB-141: [GAT]Calculating Signatures For Windows File Systems

GSA intends to make the clarifications detailed in the following pages in the next release of the GAT protocol.

The changes in this Technical Bulletin have an impact on GAT certification.

Trademarks and Copyright: Copyright © 2017 Gaming Standards Association (GSA). All trademarks used within this document are the property of their respective owners. Gaming Standards Association and the puzzle-piece GSA logo are registered trademarks and/or trademarks of the Gaming Standards Association.

This document may be copied in part or in full by members of GSA, or non-members that have been authorized by the GSA Board of Directors, provided that ALL copies must maintain the copyright, trademark and any other proprietary notices contained on/in the materials. NO material may be modified, edited or taken out of context such that its use creates a false or misleading statement or impression as to the positions, statements or actions of GSA.

Appendix C

Directory Signature Calculations

1.1 Introduction

This Appendix defines a standard method for generating signatures of files that are contained in a Windows directory structure.

1.2 File Sorting

Windows file systems are case insensitive. For this reason, all directory and file names **MUST** be converted to uppercase prior to sorting. When processing the files that make up the directory signature, the files **MUST** be sorted by directory name first and then the filename. The directory and file names **MUST** be sorted in accordance with ISO/IEC 14651:2016, International string ordering and comparison.

1.3 File Reading

Once the sorted list of files is created, the contents need to be read and hashed. This section outlines the reading and hashing process for the various types of files that are encountered on Windows FAT and NTFS file systems.

- **Regular Files:** Normal files **MUST** be hashed using the designated algorithm by reading the contents from the beginning to the end. Seeds, salts, and offsets **MUST** not be utilized in the hashing of the individual files.
- **Shortcuts:** Windows shortcuts are files that point to another file on the computer. Shortcuts **MUST** be hashed using the designated algorithm by reading the internal binary contents from the beginning to the end. Seeds, salts, and offsets **MUST** not be utilized in the hashing of the individual files.
- **System Files:** Some files are not accessible when Windows mounts a file system. When these files are encountered, the file name **MUST** be hashed instead of the contents.
- **Reparse Points:** A reparse point is an NTFS file system object that contains a reparse tag and data that is interpreted by a file system filter identified by the tag. Reparse points **MUST** be ignored.

1.4 Directory Hashing

For the purpose of calculating the directory signature, the hashing results for all individual files **MUST** be converted to their uppercase hexadecimal ASCII equivalent characters. All white space, symbols and special characters **MUST** be removed from the hashing results prior to calculating the directory signature.

The signature of a directory is a mathematically calculated value consisting of a variable number of hexadecimal digits depending on the designated algorithm. The directory signature **MUST** be calculated over a single binary buffer where the contents consist of a concatenated list of the individual file hashing results formatted as defined above and sorted as defined in Section 1.2.

1.5 Seeds/Salts/HMAC

Seeds, salts, and HMAC are used to introduce randomization into a signature. All seed, salt, and HMAC techniques utilize data input from an external source to add some unpredictability to the final signature. Seed, salt and HMAC algorithms **MUST** only be applied to the binary buffer used in the directory hashing process defined in Section 1.4.