International Journal of Cyber-Security and Digital Forensics (IJCSDF) 7(1): 79-86
The Society of Digital Information and Wireless Communications (SDIWC), 2018 ISSN: 2305-001

Audio Steganography via Cloud Services: Integrity Analysis of Hidden File

Ashar Neyaz and Cihan Varol
Department of Computer Science
Sam Houston State University
1803 Ave I, AB1 214, Huntsville, Texas, 77341, USA
axn026@shsu.edu, *cxv007@shsu.edu

ABSTRACT

Steganography is implemented by manipulating data within a carrier file, such as on audio files. Audio Steganography is one of the techniques used for transmitting hidden information by modifying or altering an audio signal in an imperceptible manner. These hidden files can be transmitted on the internet without being detected or not stored in cloud services to carry out hidden communication or agenda. In this paper, first, background, techniques, and experiments related with audio steganography are covered. Second, different formatted audio files (.au, .ape, .mp3, .wav, and .wma) will be altered to hide a hidden message via using well-known steganography tools. Third and last, this data will be transmitted through cloud and internet via services such as Facebook Messenger, iMesssage, Google Drive, DropBox, and Amazon Drive to analyze whether the files retains the hidden message throughout transmission. After conducting the experiments, it is observed that integrity of hidden data is retained when the cloud service is Google, Amazon, and Dropbox. With using Steg Hide tool, hidden message can be retained when Facebook Messenger is used to transfer the file. None of the steganography tools that were employed maintained the integrity with iMessage either because of incompatible file format or because of not being able to retrieve the hidden data. Overall, since there are differences among the algorithms and input file formats used in the cloud services, standardizing these services will help to not lose the integrity of files when storing/transferring.

KEYWORDS

Audio Steganography, Cloud Computing, Steganography

1 INTRODUCTION

Nowadays, sharing multimedia materials online is a common activity. With so many media formats are being used widely, this gives endless possibilities and places to hide information. With cloud computing in demand everywhere, people are storing and transmitting information in the cloud services on everyday basis. This technology also creates an opportunity to transfer hidden information through variety of cloud services.

Different cloud services use different algorithms to store data. People exchange different kinds of files over the internet and also save them on their cloud storage without even checking the integrity of the files or have an understanding of the storage/transmission algorithm. What if one hide and send an important information in order to save the bandwidth of their internet data, would that lead to corruption in the data? What happens to the digital signatures or electronic watermarking of the files and more importantly the integrity of the file or the message being transmitted? These vital questions need to be addressed. To answer these questions, a secret message is embedded behind different audio files (.au, .ape, .mp3, .wav, and .wma) and verify which files will be successfully transmitted through Facebook Messenger, iMessage, Google Drive, Dropbox, and Amazon Drive. After successfully transmission of the audio files, observation of whether the files retained the hidden message will determine the ultimate success of the audio files transmissions through these cloud services.

The rest of the paper is formatted as follows. Section 2 provides relevant work in audio steganography in cloud computing. Audio Steganography concept and hiding techniques are discussed in Section 3. Section 4 briefly introduce cloud storage techniques and challenges bringing to digital forensics. Methodology and the conducted experiment is detailed in Section 5. Section 6 provides details about transmitting of the audio files in the cloud services. At the end, the paper is finalized and possible future work is shared in Section 7.

2 BACKGROUND

Number of studies have been conducted in the literature for securing data in the cloud. In this section, some of the recent recent stenographic techniques that are implemented to secure the data in the cloud will be shared. Mandai and Bhattacharyya extended classic image steganograpghy technique by using another technique to pock the pixels of cover image where the hidden data will be stored. Specifically, they propose a data position scrambling PMM and genetic algorithm based secret key image encryption method [1]. Mohis and Devipriya created public key encryption scheme based on a mediated certificate-less encipherment to hide information in images. When different clients utilizes same arrangements of access control then this technique can perform encipherment only once for each information to decrease the overhead at the proprietor side [2].

Murakami et al. applied dynamically generated morphing images as cover medium to enhance the security which do not require keys to decrypt [3]. Ranjan and Bhonsle utilized AES cryptography alongside information proprietor control to internal or external clients for steganography [4].

One of the closest study to our work was conducted to see if steganography can be used and automated using Facebook messaging system [5]. In this work authors reflected that Facebook Cover Photos can effectively hide information to at least twenty percent capacity using Discrete Cosine Transform (DCT) coefficient embedding algorithms. The authors tried to develop a JavaScript Facebook Steganography application. However, they concluded that an automated, Facebook-integrated application is not desirable and manual interaction with Facebook should be used [5].

As can be reflected above, steganography, especially audio, combined with cloud storage services is relative new for exploration and not a lot of work has been done in this combined concept. Steganography and cloud storage have advantages but at the same time they have a fair share of disadvantages too. To our best knowledge, at the moment, no forensic work exist to investigate steganography and cloud service at the same time for security breaches. Therefore, this area needs to be explored since confidentiality, integrity, and availability are involved and cloud storage is still prone to security compromises.

3 AUDIO STEGANOGRAPHY

Audio steganography is a type of stealth technique that can used to hide secret information inside of an audio file. The objective of steganography is to hide the information and provide reliable transmission of the hidden information [6]. When dealing with audio steganography a different aspect of steganography must be applied. Avoiding detection, while still maintaining the resemblance to the carrier file is one of the main goals of audio steganography. There are numerous techniques and algorithms that can be used to demonstrate these techniques of audio steganography [6] [7]. In below most widely used ones are reflected.

3.1 Least Significant Bit(s)

Least Significant Bit encoding is a very common and simple technique that has been used as one of the earliest implementations when preforming audio steganography. The process of LSB can embed data image in an audio and also can embed audio data in an image file [6]. This is performed by embedding each bit from the desired hidden message in the least significant bit cover audio file.

One of the main advantages of LSB is that with this technique there is a higher rate of watermark channel bit, which means that the odds of being detected is low. Another advantage is that the LSB is simple to implement and the complexity is much lower compared to other techniques. This simplicity of this technique brings disadvantages such as low robustness [6]. It is highly unlikely to survive watermarking techniques via this technique because bits that are using LSB provides no margin for error.

3.2 Phase Encoding

Phase coding is the technique of watermarking that substitutes the first audio segment with what is called a reference phase. This is where the hidden information is stored in echo hiding [6]. This keeps intact the rest of the file by preserving the other phases between the segments. One of the main disadvantages is that there is not a lot of data that can used in phase coding because there is a low payload for embedding data. Also, with all the data being encapsulated within the first block of the audio file, this makes it easier for attackers to remove the data [6].

3.3 Echo Hiding

Echo hiding encompasses embedding data into a carrier signal by introducing a short echo. With echo hiding, there are three considerations that are utilized, which are amplitude, decay rate, and offset. Using these considerations allow the short echo to be embedded into the carrier audio file with very little detection because the audio file is edited to fit the added noise [6]. Advantages of echo hiding are mainly the simplicity of the implementation and also having embedding rate low [6]. Disadvantages of using echo hiding is that there is low security with this technique. Also, there is more complicated computations for detecting the hidden data, which might cause to take longer to detect [6].

3.4 Spread Spectrum

Spread spectrum is the last measurement to take into consideration. This is a technique that allows any steam of information to be embedded at any frequency spectrum as possible. With the simple process of spread spectrum, high robustness and security are possible when preforming this technique [6]. Disadvantages is that the transformation functions can cause long delays when applied to the inverse transform functions that are needed in order to interpret the hidden message [6].

4 CLOUD COMPUTING

National Institute of Software and Technology (NIST) defines cloud computing as, “Cloud computing is a model which provides a convenient way of on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications and services), that can be rapidly provisioned and released with minimal management effort or service provider interaction” [8]. Cloud computing comes with a lot user-beneficiary characteristics such as, resource pooling, rapid elasticity, on-demand self-service, broad network access, and measured elasticity [9][10]. There are different kinds of services and deployment methods such public cloud, private cloud, community cloud, and hybrid cloud.

The cloud computing incorporates three types of services, which are: *Software as a service (SaaS):* This cloud computing service provides users to use software applications which can be accessed through web browsers. Examples are: Google Docs, Google Drive, and Salesforce. In this service, users can only use the services provided by this model through web browsers and they have no control on its underlying infrastructure such networks or storage [9][10]. *Platform as a service (PaaS)*: In this cloud computing service, users can deploy their own software applications in the cloud environment. The operating systems are already installed, the users just have to install their own applications on it. Users still don’t have the control on underlying infrastructure such as network, storage, servers, operating systems but they do have control on their deployed applications. Examples are: Google App Engine, Windows Azure [9][10][11]. *Infrastructure as a service (IaaS):* This cloud computing service provides users to rent the hardware infrastructure like servers and workstations and users in turn can install any operating systems or software applications on it. The service itself is very scalable as the users can scale up and down their requirement according to their needs. Users have full control over operating systems, storage, deployed applications, and also have a control over few selective network components too. Example: Amazon EC2 [12].

This work particularly focuses on *SaaS* type of services. One of the important issue with cloud computing as *SaaS* is the file integrity and forensic analysis of the data located in cloud services [13]. If resources of systems are effectively segregated SaaS as this could provide

a threat against data integrity [14]. This work will shed a light on how cloud services handle hidden information embedded on audio files.

5 EXPERIMENTAL TOOLS and AUDIO FILES

Our experiment is based on testing different audio files over different audio steganography software programs to determine if the hidden audio files are capable of retaining the embedded message after transmission. This is implemented by embedding the secret message behind the audio files and verifying which files were successfully transmitted through cloud and internet services such as Facebook Messenger, iMessage, Google Drive, Dropbox and Amazon Drive. After successfully transmission of the audio files, observation of whether the files retained the hidden message will determine the ultimate success of the audio files transmissions. Before moving further and giving the details of the experiment, the paper gives an idea of the kind of files and audio steganography tools being used to carry out the experiment.

5.1 Audio File Types

5.1.1 .au file

Sun Microsystems have developed the .au audio file format. This type of audio file was very common on NeXT systems and on early Web pages. It was originally header-less, and was simply 8-bit μ-law-encoded data at an 8000 Hz sample rate. Newer versions of .au files have a header that consists of six unsigned 32-bit words, an optional information chunk and the data in big endian format [15][16]. Open Puff is able to recognize this type audio file format and embed the hidden message behind the audio file for steganography purposes.

5.1.2 .ape file

It is a file format for lossless audio data compression. It does not discard data during the process of encoding unlike the lossy compression ones. File encoded with .ape format are typically half of the original size. It is also called Monkey's Audio [17][18]. Deep Sound is able to encrypt this type of audio file.

5.1.3 .mp3 file

MP3 is a type of audio file that was developed in 1987. The advantage to using this audio file for steganography is the high compression rate of 1/11 while still maintaining a great quality, high availability for the decoders, and low CPU requirements for playback. The quality of the compression is so high that it makes it difficult to distinguish the results of MP3 range 160-224 kbps from the original material [19].

5.1.4 .wma file

Windows Media Audio or *WMA* is an audio file developed by Microsoft. It is a part of the Windows Media framework and it consists of four distinct codecs. The original WMA codec or *WMA*, *WMA Pro*, a lossless code called *WMA Lossless*, and *WMA Voice*. WMA file is compatible with Deep Sound for the process of steganography [20]. AES encryption is used by Deep Sound for the encryption of this type of file.

5.1.5 .wav file

WAVE file or *Waveform Audio File Format* (.WAV filename extension) was developed by Microsoft and IBM. It is an audio file format standard for storing an audio bitstream on computers. It uses Resource Interchange File Format (RIFF) bitstream format method for storing data in "chunks", and is also close to the 8SVX and the AIFF format used on Amiga and Macintosh based computers [21]. This format is very common in Windows computers and used for raw and typically uncompressed audio. It uses the linear pulse-code modulation (LPCM) format for usual bitstream encoding [22].

5.2 Audio Steganography Tools

5.2.1 OpenPuff

OpenPuff Steganography and Watermarking, sometimes abbreviated as OpenPuff or Puff, is a freeware steganography tool developed for Microsoft Windows systems created by Cosimo Oliboni. This program was the first steganography tool released on December 2004 that has the following features [23]:

- Allows users to hide data in more than one single carrier file,

- Implements three layers of hidden data obfuscation i.e. cryptography, whitening, and encoding
- Extends deniable cryptography into deniable steganography.

5.2.2 DeepSound

DeepSound is also a steganography tool and audio converter that hides hidden message or secret data into audio files. It also extracts secret files directly from audio files. This software is also used as copyright marking software for wave, flac, wma, ape, and audio CD file formats. DeepSound also support encrypting secret files using AES-256 (Advanced Encryption Standard) to improve data protection [24]. Furthermore, the application also contains an easy to use Audio Converter Module that can encode FLAC, MP3, WMA, WAV, and APE and interchange to each other.

5.2.3 *MP3 Stego*

MP3Stego will hide information in MP3 files during the compression process. The data is first compressed, encrypted, and then hidden in the MP3 bit stream. This can be used as watermarking marking system in mp3 files [25].

5.2.4 Steghide

Steghide is a steganography program that is able to hide data in various kinds of image and audio files. The color respective sample frequencies are not changed thus making the embedding resistant against first-order statistical tests [26].

5.2.5 Xiao Audio Steg

This is very easy to use tool that uses only two files as input, a text file and .wav sound file. It uses any kind of file as stego file to be embedded in the input files. It also uses different kinds of encryption / decryption techniques such as RC2, RC4, DES, Triple DES, and Triple Des 112 and hashing MD2, MD4, MD5, and SHA Algorithms through using password protected [27].

5.2.6 Silent Eye

Silent Eye is a cross-platform software designed for an easy use of steganography, like hiding messages into pictures or sounds. It has a nice GUI and has an easy integration of new steganography algorithm and cryptography process by using a plug-ins system. Silent Eye is free to use (under GNU GPL v3) [28]. The main features of the software are:

- Hide information into images and sounds by using LSB Uses AES 128/256 encryption techniques for the data
- Capacity to hide text or file, zlib compression of message, Drag & Drop

The following table (Table 1) shows the audio steganography software with audio files compatibility.

Table 1. File Types and Software Compability

File Types/ Software Compatibility	DeepSound	OpenPuff	MP3 Stego	Steg Hide	Xiao Audio Steg	SilentEye
.au	Non-Compatible	Compatible	Non-Compatible as input	Compatible	Non-Compatible	Non-Compatible
.ape	Compatible	Non-Compatible	Non-Compatible as input	Non-Compatible	Non-Compatible	Non-Compatible
.mp3	Compatible	Compatible	Input and output file can't be same.	Non-Compatible	Non-Compatible	Non-Compatible
.wav	Compatible	Non-Compatible	Takes .wav as input file	Compatible	Compatible	Compatible
.wma	Compatible	Non-Compatible	Non-Compatible as input	Non-Compatible	Non-Compatible	Non-Compatible

6 EXPERIMENTAL RESULTS

In order to transfer and store files to check for integrity, cloud services that are widely known are used, such as Facebook Messenger, Google Drive, iMessage, Dropbox, and Amazon Drive. Both decoding and encoding is done with the same steganopgrahy tools after the audios are transferred through cloud services. As reflected in Table 2, with DeepSound, OpenPuff,

MP3Stego, Xiao Audi Steg, and SilentEye integrity of data is lost when the file was uploaded on Facebook Messenger and iMessage but Google, Amazon, and Dropbox were able to maintain the integrity of data. Steg Hide goes beyond these tools above and although it puts the audio for an infinite loop still the hidden information were able to get retrieved when using Facebook Messenger. None of the tools that were used were able to maintain the integrity with iMessage either because of incompatible file format or because of not able to retrieve the hidden data.

7 CONCLUSION AND FUTURE WORK

Audio Steganography is a method that is used to hide communication via embedding secret message in an audio file. Audio files are viewed as one of the most efficient methods of hiding data because of file sizes. In this work the question that was answered was whether the integrity of messages can be intact while employing cloud and internet services like Google Drive, Dropbox, Facebook Messenger, Amazon Drive, and iMessage. The purpose of the experiment was to check for the file integrity and corruption in the data since each service mentioned above are employing different algorithms when storing the dataset.

To achieve the task of hiding data behind the audio files MP3Stego, Steghide, Xiao Steganography, OpenPuff, DeepSound, and SilentEye are used. All of the software are capable of hiding secret message in audio files. It is observed that StegHide is the most compatible software when hiding messages and transferring those using cloud services. Since there is inconsistency among the algorithms used in these cloud services and limitations on the input file formats, there is a need to standardize these to transfer files without losing their integrity.

As a future work, this can be extended to test how these cloud services react to different file formats carrying a secret message, such as videos, images, documents, etc.

Table 2. File Types and Software Compatibility

Tool	Files Format	Facebook Messenger	Google Drive	iMessage	Dropbox	Amazon Drive
DeepSound	.wav .flac .ape	**Unsuccessful** Accepted but converted into mp4 format and corrupted the hidden message.	**Success**	**Unsuccessful** Not acceptable as input file	**Success**	**Success**
OpenPuff	.wav	**Unsuccessful** Accepted but converted into mp4 format and got corrupted. Looping forever in the messenger	**Success**	**Unsuccessful** Not acceptable as input file	**Success**	**Success**
	.mp3	**Unsuccessful** Accepted but converted into mp4 format and corrupted the hidden data	**Success**	**Unsuccessful** But hidden file cannot be extracted	**Success**	**Success**
MP3 Stego	.wav	**Unsuccessful** While uploading, file format changed and hidden data was lost	**Success**	**Unsuccessful** Not acceptable as input file	**Success**	**Success**
Steg Hide	.au	**Success** But loops around forever when played on the messenger	**Success**	**Unsuccessful** Converted into mp3 and did not play.	**Success**	**Success**
	.wav	**Success** But loops around forever when played on the messenger	**Success**	**Unsuccessful** Not acceptable as input file	**Success**	**Success**
Xiao Audio Steg	.wav	**Unsuccessful** While uploading, file format changed and hidden data was lost	**Success**	**Unsuccessful** Not acceptable as input file	**Success**	**Success**
SilentEye	.wav,	**Unsuccessful** Accepted but converted into mp4 format and got corrupted. Also, size got considerably reduced	**Success**	**Unsuccessful** Not acceptable as input file	**Success**	**Success**

REFERENCES

1. Mandai, S., Bhattacharyya, S.: Secret Data Sharing in Cloud Environment Using Steganography and Encryption Using GA. 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), pp. 1469–1474. https://doi.org/10.1109/ICGCIoT.2015.7380699 (2015).
2. Mohis, M., Devipriya, V.S.: An improved approach for Enhancing Public Cloud Data Security through Steganographic Technique. IEEE In Inventive Computation Technologies (ICICT). (2016).
3. Murakami, K., Hanyu, R., Zhao, Q., Kaneda, Y.: Improvement of security in cloud systems based on steganography. 2013 International Joint Conference on Awareness Science and Technology & Ubi-Media Computing (iCAST 2013 & UMEDIA 2013), pp. 503–508. https://doi.org/10.1109/ICAwST.2013.6765492 (2013).
4. Ranjan, A., Bhonsle, M.: Advanced technics to shared & protect cloud data using multilayer steganography and cryptography. IEEE 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), pp 35–41. (2016).
5. Amsden, N., Chen, L., Yuan, X.: Transmitting Hidden Information Using Steganography via Facebook. IEEE 2014 Conference on Computing, Communication, and Network Technologies (ICCCNT 2014), doi: 10.1109/ICCCNT.2014.6963080. (2014).
6. Binny, A., Koilakuntla, M.: Hiding Secret Information Using LSB Based Audio Steganography. 2014 International Conference on Soft Computing & Machine Intelligence. doi: 10.1109/ISCMI.2014.24. (2014).
7. Taneja, N., Gupta, P.: Implementation of Dual Security through DSA and Audio Steganography. 2015 International Conference on Green Computing and Internet of Things (ICGCIoT 2015). (2015).
8. Zawoad, S., Hasan, R.: Cloud forensics: A meta-study of challenges, approaches, and open problems. arXiv:1302.6312v1. (2013).
9. Dykstra, J., Sherman A.T.: Understanding issues in cloud forensics- two hypothetical case studies. 2011 Proceedings of the Conference on Digital Forensics, Security, and Law. (2011).
10. Puthal, D., Mishra, S., Swain S.: Cloud Computing Features, Issues, and Challenges: A Big Picture. 2015 International Conference on Computational Intelligence and Networks (CINE 2015). Doi: 10.1109/CINE.2015.31. (2015).
11. Sang, T.: A log-based approach to make digital forensics easier on cloud computing. 2013 Third International Conference on Intelligent System Design and Engineering Applications. doi: 10.1109/ISDEA.2012.29. (2013).
12. Nelson, B., Philips, A., Steuart C.: Guide to Computer Forensics and Investigation. Fifth Edition, pg 483-484. (2009).
13. Daryabar, F., Dehghantanha, A., Udzir, N.I., Sani, N.F.M, Shamsuddin, S., Norouzizadeh, F.: A Survey About Impacts of Cloud Computing on Digital Forensics. International Journal of Cyber-Security and Digital Forensics (IJCSDF), 2. pp. 77-94. (2013).
14. Keat Y.S., Rad, B.B., Ahmadi, M.: Cloud Computing Security and Forensics Issues and Awareness of Cloud Storage Users in Malaysia. International Journal of Cyber-Security and Digital Forensics (IJCSDF), 6. pp. 1-13. (2017).
15. Bourke, P. (September 1996). Creating AIFF Audio Formatted Files. Retrieved from: https://en.wikipedia.org/wiki/Au_file_format
16. .Au File Extension. Retrieved from: https://www.reviversoft.com/file-extensions/au
17. What is APE?. Retrieved from: http://www.coolutils.com/Formats/APE
18. Ashland, M. Monkey's Audio. Retrieved from: http://www.monkeysaudio.com/
19. Sterne J: MP3: The Meaning of a Format. Duke University Press Books. (2012).
20. WMA File Format. Retrieved from: http://whatis.techtarget.com/fileformat/WMA-Audio-file-in-Microsoft-Windows-Media-format
21. Sapp, C.S. WAVE PCM soundfile format. Retrieved from: http://soundfile.sapp.org/doc/WaveFormat/
22. Mingguang Z., Zhitang, L. A wav-audio steganography algorithm based on amplitude modifying. 2014 10th International Conference on Computational Intelligence and Security. doi: 10.1109/CIS.2014.78. (2014).
23. OpenPuff (2008). Embedded SW. Retrieved from: http://embeddedsw.net/OpenPuff_Steganography_Home.html
24. Batora, J. (2015 November). DeepSound Overview. Retrieved from: http://jpinsoft.net/deepsound
25. Petitcolas F. (2006 June 13). MP3Stego. Retrieved from: http://www.petitcolas.net/steganography/mp3stego/
26. Hetzl S. (2003 October 9). Steghide. Retrieved from: http://steghide.sourceforge.net/
27. Softsonic. (2015). Xiao Steganography. Retrieved from: https://xiao-steganography.en.softonic.com/
28. SilentEye. (2016). What is Silent Eye?. Retrieved from: http://silenteye.v1kings.io/