A Statistical Approach to the Detection of HTML Attribute Permutation Steganography

Sedeeq Iman, Coenen Frans and Lisitsa Alexei
Department of Computer Science, University of Liverpool, Liverpool, U.K.

Keywords: Steganography, Standard Deviation, Attribute Permutation.

Abstract: A mechanism for monitoring WWW pages to identify the presence (or otherwise) of attribute permutation steganography is presented. The proposed mechanism is based on a statistical approach, more specifically an attribute position Standard Deviation (SD) measure is used to detect the modification of attribute locations within webpages. The monitoring of web pages involves usage of a SD threshold. The determination of this threshold involves a training process using seeded training data. Once a threshold has been learnt monitoring can be commenced. An evaluation of the process is presented indicating that the attribute position SD concept can be successfully used to monitor web pages for attribute permutation steganography.

1 INTRODUCTION

Steganography is the art of hiding messages within an innocent cover carrier such as digital media (image, audio, video, text) in such a way that casual observers would not be aware of the existence of the hidden message, only the sender and the intended receiver. New networking environments, such as cloud computing and wireless networks environments, and new communication devices such as smart phones have also created many new opportunities for steganographers(Wendzel et al., 2014; Zielińska et al., 2014).

The increasing use of social media has facilitated a huge increase in the amount of information exchanged over the internet, and consequently concerns about the threat of steganography have increased correspondingly. As a result there is a significant need for steganographic countermeasures; techniques and processes to determine whether a file (*carrier*) includes a hidden message or not, and if so techniques and processes to extract this secret message and/or destroy it.

For steganography to operate the following characteristics of the cover medium are important: (i) be a common (popular) medium (such as web pages or email) to avoid raising suspicion, and (ii) allow changes to be made without this being noticeable to casual observers. HTML encoded web pages are an ideal cover medium. HTML is the language of the web; HTML documents are thus the most popular file format used to present information over the Internet. Also HTML files have many enriched features like scripts and hyperlinks that can be exploited when wishing to embed secret messages. HTML encoded web pages thus feature the necessary cover medium characteristics identified above. Consequently HTML encoded web pages have attracted the attention of steganographers as the cover carrier of choice.

This paper presents an approach to the detection of hidden messages that are embedded using what is known as “attribute permutation” steganography. The fundamental idea presented is that the monitoring the Standard Deviation (SD) of attributes locations within a webpage can be used to detect the presence of hidden message. The work is based on the conjecture that the SD increases when attribute permutation steganography takes place. We provide the empirical evidence that indeed the conjecture holds true. In order to use it for the steganography detection though one has to take into account that (1) SD computed for different HTML encoded web pages without any embedded messages may vary significantly; (2) SD computed for the same web page with dynamic content may vary from time to time. We propose a monitoring procedure which when applied for a web page with dynamic content, learns a *normal* level of SD and its *threshold* value, any excess of which is interpreted as the presence of a hidden message.

Iman, S., Frans, C. and Alexei, L.
A Statistical Approach to the Detection of HTML Attribute Permutation Steganography.
DOI: 10.5220/0005801705220527
In *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP 2016)*, pages 522-527
ISBN: 978-989-758-167-0

2 HTML STEGANOGRAPHY

HTML (Hyper-Text Markup Language) comprises markup tags which tell a browser the format in which to display tables, paragraphs, images and so on. Most tags are paired using a start tag and an end tag, with text content in between. According to the HTML specification tags can also have attributes that can be used to customize the tag, and these attributes have values assigned to them by default or by the user. The HTML specification does not determine the order of these attributes. Moreover HTML allows for the inclusion of hyperlinks and scripts; enriching features that represent a fertile ground for steganography. Steganographic techniques in HTML can be largely classified into three groups: (i) Embedding invisible characters (Sui and Luo, 2004), (ii) Switching tag letters case (Zhao and Lu, 2007) and (iii) Attribute permutation (S.Forrest, 2006; Huang et al., 2008; Shen and Zhao, 2010). The latter is the focus of the work presented in this paper.

2.1 Attribute Permutation

Attributes in HTML tags can come in any order, the HTML specification does not prescribe any attribute ordering. Therefore a tag with eight attributes has $8! = 40320$ equivalent attribute configurations that are all processed in an identical manner by a browser. The ordering of the attributes can be used to embed a hidden message.

From the perspective of the steganographer attribute permutation offers the same advantages as the switching tag letter case method: (i) the presentation of the HTML file in a browser gives no indication of there being any hidden messages and (ii) the file size is unaffected. However, in addition, it is difficult to identify by inspection of the HTML source code.

Several algorithms have been proposed to achieve attribute permutation steganography, examples can be found in (S.Forrest, 2006; Huang et al., 2008; Shen and Zhao, 2010). The method used with respect to the evaluation presented in this paper is that of (Shen and Zhao, 2010) where a relation between tag attributes is defined using a binary string which is later transformed into a set of permutations and the current tag is randomly replaced with one of these permutations.

The attribute permutation embedding capacity of a given web page is defined as the number of bits of secret message that we can embed in a given webpage H. According to the steganography algorithm in (Shen and Zhao, 2010) we can embed $(n-1)$ bits of secret message in a webpage using tags that have more than one attribute. The Maximum Embedding Capacity (MEC) in bytes can be calculated using equation 1:

$$MEC(H) = \frac{1}{8} \sum_{T_j \in Q} (|T_j| - 1) \qquad (1)$$

where Q is the set of tags featured in H of two or more attributes, and $|T_j|$ is the number of attributes in tag T_j, the jth tag in Q. We multiply by $\frac{1}{8}$ so that the result is in bytes.

3 STANDARD DEVIATION OF ATTRIBUTE LOCATIONS

The main idea presented in this paper is to identify the presence of attribute permutation steganography by monitoring the Standard Deviation (SD) of attributes locations within a webpage. This is based on the assumption that for the HTML pages without message embeddings the attributes tend to be located in the same locations for the most of the tags. By reordering attributes in tags, so as to hide a message, the *coherency of the attribute locations* changes. More specifically the dispersion of attribute locations will increase if message hiding is taking place, in other words the variance of attribute locations will increase. The conjecture is that the above can be usefully employed to distinguish a stego file from a regular file.

Before presenting the proposed stego file identification algorithm the following definitions should be noted:

- H is a webpage.
- Q: is a set of tags in H that have two attributes or more, $Q = \{T_1, T_2, \ldots\}$.
- $T_j = \{a_1, a_2, \ldots, a_n\}$ denotes the jth tag in Q with its attributes.
- $|T_j|$: represents the number of attributes in the jth tag.
- $ANS = \{a_1, a_2, ...a_n\}$: is the complete set of attributes in Q.
- $ALS(a_i)$: A set of locations in Q for attribute a_i. $ALS(a_i) = \{la_{i1}, la_{i1}, \ldots la_{im}\}$. Each attribute has m occurrences such that : $|ALS(a_i)| = m$.
- $VALS(a_i)$: is the variance of $ALS(a_i)$:

$$VALS(a_i) = \frac{\sum_{k=1}^{m} (la_{ik} - av)^2}{m} \qquad (2)$$

$$av = \frac{\sum_{k=1}^{m} la_{ik}}{m} \qquad (3)$$

- VQ: is the total variance of all attributes in Q. The total variance is equal to the sum of their variances as shown in equation 4 below:

$$VQ = VAR(\sum_{i=1}^{m} VALS(a_i)) = \sum_{i=1}^{m} VALS(a_i) \quad (4)$$

- SD: is the standard deviation of VQ in order to quantify the amount of variation of a set of a web-page attribute locations. It can be calculated by:

$$SD = \sqrt{VQ} \quad (5)$$

The process for calculating the SD of attributes is given in Algorithm 1. The algorithm operates as follows: The input is a webpage H. The output is the standard deviation of the set of Q attributes locations.

Algorithm 1: Calculate standard deviation SD of attributes locations of a webpage.

```
inputH
ANS = {}
ALS = {}
Find Q
for each tag T ∈ Q do
    for each attribute a ∈ T do
        ANS = ANS + a.name
        ALS = ALS + a.location
    end for
end for
for each attribute a ∈ ANS do
    for i = 1 to i = m do
        VALS(a_i) = (Σ_{k=1}^{m} (la_ik − av)^2) / m
        av = (Σ_{k=1}^{m} la_ik) / m
    end for
end for
VQ = Σ_{i=1}^{m} VALS(a_i)
SD = √VQ
```

4 ATTRIBUTE PERMUTATION STEGANOGRAPHY MONITORING PROCESS

Using the above approach we can determine the attribute location SD for a given www page H. By monitoring the attribute location SD of H over a period of time the conjecture is that we can identify attribute permutation steganography whenever unusual SD values are detected. This requires the establishment of a webpage steganography detection threshold σ. The idea is that the value for σ can be learnt by deliberately seeding a sequence of "snapshots" $C = \{s_1, s_2, \ldots, _n\}$of H with hidden messages (using attribute permutation steganography). In other words the monitoring commences with a "training process" during which the value for σ is learnt. More specifically snapshots of the web page of interest are collected periodically at intervals of time τ over a period of time T. For each snapshot a message of length L is generated ($L \leq MEC(H)$) and the SD before and after message embedding calculated. The value for σ is then calculated using equation

$$\sigma = (\alpha \times SD_{avbefore}) + ((1-\alpha) \times SD_{avafter}) \quad (6)$$

Where: (i) α is a user specified sensitivity factor of between 0.1 and 1 (ii) $SD_{avbefore}$ is the average SD without message hiding and (iii) $SD_{avafter}$ is the average SD with message hiding. Note that a high α value will have the effect of reducing σ and making the detection process more sensitive. Note that the chance of identifying false positives increases as the value for σ decreases.

The arguments of the monitoring process are as follows:

- H: is (HTML encoded) web page with dynamic content.
- α: is a sensitivity factor, $0 < \alpha < 1$.
- T: is the total training time.
- τ: is the interval to capture a webpage snapshot periodically for example every two hours.
- L: is the length of embedded message.

The actual monitoring process then comprises three steps:

1. Collecting snapshots of H with interval τ over time period T to give the set $C = \{s_1, s_2, \ldots, s_n\}$
2. Calculating the threshold σ .
   (a) For each $s_i \in C$ compute the SD of snapshot s_i, then embed a message using the algorithm in (Shen and Zhao, 2010) and compute the new SD.
   (b) Compute the average of the SDs before embedding ($SD_{avbefore} = \frac{\sum_{i=0}^{k} SD_i}{k}$) and the average of the SDs after embedding ($SD_{avafter} = \frac{\sum_{i=0}^{k} SD_i}{k}$); Where k represents the number of snapshots.
   (c) The σ value is then calculated using equation 6.
3. Detecting: compute SD of a suspected instance of H and compare it with σ; If SD$\geq \sigma$ then the page is a stego page otherwise the page is an unmodified page.

5 EVALUATION

The objectives of the evaluation of the proposed approach were as follows:

1. To confirm that the conjecture that increases in the attribute locations SD of webpages is indeed an indicator attribute permutation steganography.
2. To compare naturally occurring changes in SD values caused by the dynamic content with the changes caused by message embedding.
3. To test how the type of the embedded text may affect the changes in SD.
4. To conform that the attribute locations SD concept could be effectively used to monitor web pages for attribute permutation steganography.

5.1 Attribute Locations SD as an Indicator of Attribute Permutation Steganography

The experiments designed to establish that the attribute locations SD of webpages could indeed be used to detect attribute permutation steganography were founded on three "snapshots" of the landing page of three well known websites: (i) Sony (ii) BBC and (iii) Wikipedia. In each case attribute permutation was used to embed hidden message using the technique presented in (Shen and Zhao, 2010) (as described previously in Section 2). Messages of different length L were embedded from between 10% to 100% of the MEC, incrementing in steps of 10%. The MEC for the BBC, Wikipedia, Sony pages were 70 bytes, 74 bytes and 117 bytes respectively.

The results are as shown in Figure 1. The x-axis lists the message size as a percentage of the MEC for each landing page; the y-axis then gives the SD. From the figure it can be observed that the SD starts to increase, with respect to the SD for the landing page without any hidden message embedding, as the size of the embedded message increases. When the size of the hidden message was equivalent to the MEC for each given landing page the recorded increase in SD equated to 13.1%, 14.6% and 21% respectively. Thus, from the above, we can conclude that attributes location SD can indeed be used for the purpose of identifying attribute permutation steganography.

5.2 Natural Changes in SD vs Changes Caused by Embedding

To compare naturally occuring changes in SD values caused by the dynamic content with the changes

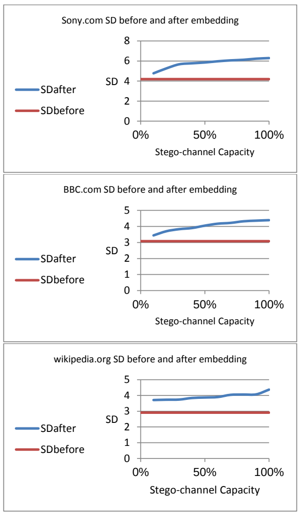

Figure 1: Standard deviation SD of attributes locations dependency on the message length.

caused by message embedding the following experiments have been done.

Ten sequences of snapshots (40 snapshots per sequence) were used based on the landing pages for ten well known web sites covering a range of www domains (news, education, company, shopping): (i) BBC (ii) New York Times (iii) Wikipedia, (iv) Stackoverflow (v) Sony (vi) The University of Liverpool (vii) IEEE (viii) WebMD (x) Microsoft and (x) Amazon. For collection purposes the following parameters were used: (i) $T =$ one week and (ii) $\tau =$ every two hours. The length of embedded message was $L = 40\%$ of a webpage MEC. The SD value was calculated before and after message hiding. The results are presented in Figure 2. From the figure it can be seen that in every case changes in SD values caused by the message embedding and the changes occurring naturally are significantly different.

5.3 Various Types of Embedded Messages

The goal of the next series of experiments was to test how the type of the embedded text may affect the changes in SD. The same ten sequences of snapshots were used and the message embedding was conducted with 29 different types of messages (natural language, random letters, mixture of letters, symbols and numbers). The results including the mean of the standard deviation after embedding these 29 instances of these message samples and the standard deviation of their SDs are presented in Table 3. As we can notice from Table 3 the type of a hidden message has no differentiable effect on SD changes.

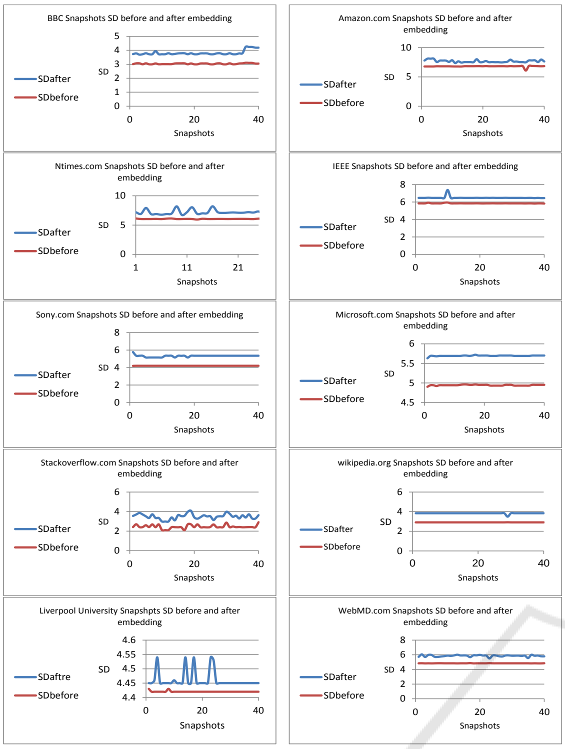

Figure 2: Standard deviation SD values before and after message embedding.

	Length of embedded message relative to a webpage MEC									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Mean	6.05	6.14	6.34	6.52	6.65	6.81	6.83	6.91	6.99	7.16
SD	0.04	0.06	0.10	0.12	0.11	0.12	0.12	0.16	0.15	0.17

Figure 3: Mean of Standard deviation of 29 instances of embedded messages and their SD.

5.4 Evaluation of Monitoring Process

To evaluate the proposed monitoring process the same ten sequences of snapshots used for the foregoing experiment were used within each case the first thirty allocated to training and the last ten to testing. Message embedding for evaluation purposes was conducted using the same attribute permutation algorithm from (Shen and Zhao, 2010). The messages were in natural language. Once training was completed we could go on to test the monitoring process. The testing was conducted by embedding hidden messages in natural language, using attribute permutation steganography of increasing length in each of the ten web page test sequences starting with $L = 10\%$ and increasing to $L = 100\%$ in steps of 10%. The results are presented in graphs 4 and 5 for IEEE webpage.

- Graph 4 demonstrated the detection threshold σ sensitivity of α selection. We tried different α values to get σ while L the length of embedded message is fixed to 40% of a webpage MEC. The x-axis represented the length of detected message relative to a webpage MEC while the y-axis represented the SD values of a tested webpage with σ values. As illustrated in Figure 4 σ decreased from 6.43 to 6.17 as α went up from 0.1 to 0.5. Decreasing σ means increasing of false positives.

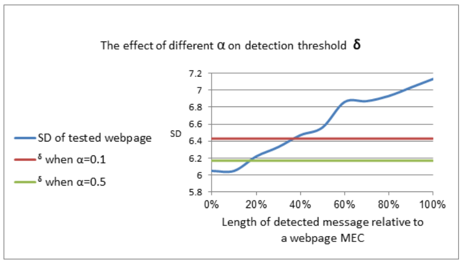

Figure 4: Detection threshold σ sensitivity of different values of α when $L = 40\%$ of a webpage MEC.

- Graph 5 demonstrated the detection threshold σ sensitivity of L selection. We tried different L values while α is fixed to 0.5 . The x-axis represented the length of detected message relative to a webpage MEC while the y-axis represented the SD values of a tested webpage with σ values of different L. As shown in Figure 5 when L became longer from 40% of webpage MEC to 60% and α is fixed to 0.5, σ increased from 6.23 to 6.48 and that would miss stego webpages with hidden messages of 40% of a webpage MEC.

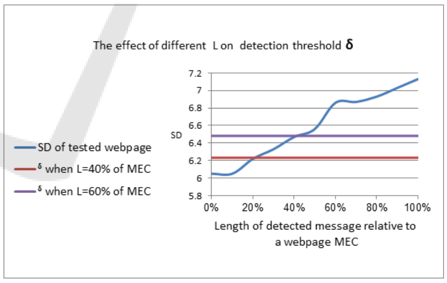

Figure 5: Detection threshold σ sensitivity of different values of L while $\alpha = 0.5$.

6 RELATED WORK AND CONCLUSIONS

In this paper we propose a process for monitoring for attribute permutation steganography in HTML encoded web pages. During the training phase the web page is sampled over a period of time and on each occasion is seeded with a hidden message. The SD is calculated for the web page with and without the hidden message. The average SDs are then used to calculate a threshold value σ which can then be used for further monitoring. The evaluation indicated that the conjecture that SD can be used for detecting attribute permutation steganography was correct.

The paper (L.Polak and Z.Kotulski, 2010) introduced an algorithm for attribute permutation steganography detection using a different statistical measure W which is based on the concept of predominant order between the pairs of attributes, it grows linearly with the occupancy of the stego channel and similarly to our method can be used for threshold based detection. The authors of (L.Polak and Z.Kotulski, 2010) discuss the issue of stego detection in the web pages with dynamic content and speculate that the detection may take into account the dynamic behavior of W: for the page containing stego messages W should either have constant high value, or fluctuate heavily. On the other hand, natural changes should be regular and periodical. No specific procedure to distinguish these cases was proposed though. We notice that our monitoring procedure can be easily deployed with computing W instead of SD.

The paper (W.Jian-feng et al., 2014) proposes the detection method for attribute permutation steganography utilizing the statistics measures and SVM classification. English translation of (W.Jian-feng et al., 2014) is not available to us at the moment, but based in particular on Fig. 1 from the paper, one may conclude that the proposed method includes computing two statistical measures of HTML page: mean of attribute positions and variance of attribute positions, which are both used as the inputs to a SVM classifier. The paper presents the experimental results on detection rates varying between 72.4% and 84.6% but it is not clear for us what was exactly the setting and whether the dependencies on the channel occupancy have been addressed.

Future work includes the evaluation of the proposed monitoring procedure for the detection the messages embedded by other steganographic algorithms utilizing attribute permutations and its comparison with the procedures from (L.Polak and Z.Kotulski, 2010) and (W.Jian-feng et al., 2014) on the same set of steganographic algorithms.

REFERENCES

Huang, H., Zhong, S., and Sun, X. (2008). An algorithm of webpage information hiding based on attributes permutation. In *4th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2008)*, pages 257–260.

L.Polak and Z.Kotulski (2010). Sending hidden data through www pages detection and prevention. *Engng.Trans.*, 58:75–89.

S.Forrest (2006). Introduction to deogol. http://www.wandership.ca/projects/deogol.

Shen, D. and Zhao, H. (2010). A novel scheme of webpage information hiding based on attributes. In *Information Theory and Information Security (ICITIS), 2010 IEEE International Conference on*, pages 1147–1150.

Sui, X.-G. and Luo, H. (2004). A new steganography method based on hypertext. In *Radio Science Conference, 2004. Proceedings. 2004 Asia-Pacific*, pages 181–184.

Wendzel, S., Mazurczyk, W., Caviglione, L., and Meier, M. (2014). Hidden and uncontrolled - on the emergence of network steganographic threats. *CoRR*, abs/1407.2029.

W.Jian-feng, H.Liu-sheng, Miao-miao, T., C.Zhi-li, and M.Hai-bo (2014). Detection of html steganography based on statistics and svm. *Journal of Chinese Computer Systems*, 35(6):1221–1225.

Zhao, Q. and Lu, H. (2007). Pca-based web page watermarking. *Pattern Recogn.*, 40(4):1334–1341.

Zielińska, E., Mazurczyk, W., and Szczypiorski, K. (2014). Trends in steganography. *Commun. ACM*, 57(3):86–95.