2012

Webpage Source Based Covert Channel

Tarun Madiraju

Daryl Johnson

Bo Yuan

Peter Lutz

Webpage Source Based Covert Channel

Tarun Madiraju, Daryl Johnson, Bo Yuan, Peter Lutz
B. Thomas Golisano College of Computing & Information Sciences
Rochester Institute of Technology, Rochester, NY USA

Abstract - *Covert Channels can be used for enabling hidden communication mechanisms that can facilitate secret message transfer. This paper presents a new covert channel based on the HTML source of a webpage. The new covert channel while featuring high bandwidth also demonstrates high imperceptibility as it doesn't involve any modifications to the source or the visibility of the webpage and is independent of timing of page requests. The availability of page source for a webpage on the Internet makes this covert channel easy to implement and effective.*

Keywords: Network Security, Covert Channels, Information Hiding, Webpage

1 Introduction

Covert channel were first defined by Lampson as a communication channel that was not intended for information transfer [1]. This covert channel, defined in 1973, was based on trusted host systems but can be applied to any shared environment. Covert channel research has migrated into many other environments since then such as networked services.

The DoD's Trusted Computer System Evaluation Criteria (TCSEC) [2] describes a covert channel as a communication channel that facilitates transfer of information between entities in such a manner that the system's security policy is violated. They also mention that the presence of a communication channel cannot be supported without the complete knowledge of the channel. A relatively recent definition states that a covert channel can be established by hiding and transmitting data over a legitimate communication channel [3].

Covert channels can be classified into three types: Storage Channels, Timing Channels and Behavioral Channels [4]. Covert storage channels include communicating entities that are involved in a direct or indirect form of writing to/reading from a storage location. Covert timing channels are described as scenarios where a process signals information to another process by modulating its own use of system resources in such a manner that the real response time observed by the second process is affected [2]. Behavioral type of channels are generally application specific wherein the behavior of an application is used to define and communicate information between the participating entities of the covert channel [4].

A typical HTML page on the Internet consists of HTML tags, regular text and URLs linking to other pages. In order to implement our covert channel, we utilize the text and the URLs available in the HTML source of a webpage. The fact that an existing webpage's source can be used to build a message dictionary without modifying any content illustrates the advantage this covert channel possesses. The purpose of this project was to demonstrate a practical covert channel that is difficult to be detected by an active administrator or a regular observer. Further, the administrator or the observer should not be able to understand the covert message if discovered.

2 Paper Organization

The remainder of this paper is organized as follows. Section 3 discusses the literature review which mentions existing work in HTML based covert channels. Section 4 details the design, methodology and experiments. Section 5 presents the characteristics of the proposed covert channel. Section 6 discusses the future work. Finally, Section 7 presents the summary of this new covert channel.

3 Literature Review

Previous works have discussed and implemented covert channels based on HTML. These works include embedding invisible characters into the html source, using uppercase and lowercase of letters in html tags, using attributes of html elements, etc [5], [6], [7], [8]. The method that involves embedding invisible characters into html source denotes white space as 0 and tab as 1. The invisible characters are embedded into the webpage by adding white spaces and tabs at the end of sentences or lines [6]. This method is simple to implement however, issues associated with this methods include: high perceptibility as a result of increase in webpage size, need to modify the web page source, and it is easier to destroy the steganographic covert channel by deleting additional white spaces and tabs. The second method is implemented by switching the case of the letters in html elements called tags [5]. The secret message was embedded by utilizing the letter case where uppercase letters were used to

represent the binary value 1 and lowercase letters were referred to as 0. This method doesn't lengthen the size of a webpage but the switched case of the letters can be used to decode the embedded message. This covert channel can be broken by converting all the tag letters to uppercase or lowercase format. The third method takes advantage of the attributes of the HTML elements. The attributes of an element in HTML can be used in any order without effecting the webpage content and visibility. The authors Dongsheng Shen and Hong Zhao used a mapping scheme between permutations of attributes and binary strings to implement an embedding and extracting process thereby achieving information hiding [7]. This method doesn't increase the webpage size and is relatively robust. Another attribute based webpage information hiding scheme described in [8] makes use of the property "equal attribute object has identical function" to define a set of rules for embedding and extracting secret information.

The following covert channel mechanisms closely relate to our proposed covert channel. William Huba et al. briefly discuss a covert channel based on the webpage browsing pattern [9] which is currently unimplemented. Erik Brown et al. presented three techniques to implement covert channels using webpage browsing patterns [10]. In the first technique, a value 0 or 1 is assigned for a page request on the basis of its distance from the root node. In the second mechanism, every page of the web site is designated as 0 or 1. The third mechanism denotes values 0 if the pattern of the page requests were in breadth-first search format while a depth-first search would indicate the value 1.

The discussed mechanisms offer low bandwidth and hence do not facilitate high capacity channels. Our approach demonstrates the characteristics of imperceptibility and robustness without compromising on the channel capacity. Unlike the first three studies mentioned above which implement unidirectional channel from the web server to the client, our work focuses on establishing a unidirectional communication channel from client to web server. In addition it focuses on using words available in the webpage as message components and URLs available in the webpage for enabling covert transfer of the message without any modifications to the existing source of web pages.

4 Proposed Covert Channel

4.1 Design

The proposed covert channel establishes a unidirectional communication scheme between the communicating entities. Typically, the communicating entities of this covert channel are a web server, as receiver, and a web client, as the sender. For any covert channel to be implemented successfully, certain information must be shared between the communicating parties prior to transferring of covert messages. In this scenario, the sending entity and the receiving entity share three things: a webpage, sender's (web client) IP and the encoding mechanism. It is important to understand that in the current proposal the receiver is implemented at the web server and hence the receiving entity must own the web server or must have compromised the web server. But the covert channel can be implemented beyond the mentioned scenario. For example, the covert channel can be implemented for scenarios where the receiver exists between the web client and the web server. This man-in-the-middle receiver captures HTTP requests from the sender in order to decode the messages based on the requested pages. An advantage of this approach is that the receiver need not have control over a web server and any webpage could be selected rather than a webpage from the owned or compromised server. This overcomes the limitation of selecting a webpage from a web server that is either owned or compromised by the receiving entity. However, in this paper we restrict our study to owned/compromised web server as the receiver.

4.2 Methodology and Experiments

The methodology section details how the encoding and decoding operations are performed by the sender and the receiver respectively. In order to present the feasibility and practicality of our webpage source based covert channel, a proof of concept was developed in python. The implementation includes two scripts wherein one of the scripts is used to carry out encoding on the web client while the other is responsible for decoding the message at the web server.

1) Sender Side: The sender side implementation of this covert channel can be described as the following phases.

Phase 1: URL Dictionary: The first task is to obtain and read and parse the webpage source. We build the URL dictionary, as illustrated in Fig. 1, by assigning numbers (termed as URL numbers) to a uniquely sorted list of URLs identified from the webpage source. The URL numbers act as keys and the URLs represent their values. If there are n URLs retrieved from the webpage, every URL is mapped to a number starting from 0 to n-1. We reserve the first three URLs, termed as Control URLs, to define activities such as indication of start or end of a covert message transfer. The purpose of each Control URL is presented in Table 1.

TABLE I
CONTROL URLS

URL Number	Indication
0	Start of communication
1	Wait for next URL
2	End of communication
3 to N-1	A word or a shift

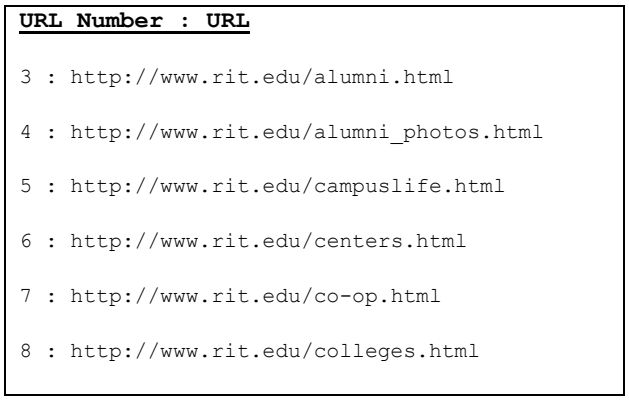

Figure 1. A sample from URL Dictionary built using RIT's homepage

Phase 2: Word List & Mapping: In addition to the retrieval of URLs, we also retrieve words from the webpage source and uniquely sort the list of words in alphabetical order. The word dictionary is obtained by mapping words from this word list to the URL number(s) as demonstrated in Fig. 2 and Fig. 3.

It is likely that in a webpage, the number of URLs would be less than the number of words in most cases, so we use the following approach to map the words and URLs. The purpose of linking URLs to words is to send or receive a word of the message based on the URL requests. Tthe URL numbers {0}, {1} and {2} are reserved for representing control URLs so we begin mapping the words from URL number {3}.. We make use of Single URL mapping and Multiple URL mapping techniques to link URLs and words. In the Single URL mapping technique, a word is linked directly to only one URL while in the Multiple URL Mapping technique a word is linked to a sequence of URLs. Starting alphabetically, the words are assigned single URLs from {3} through {n-1} to indicate the word desired. Then we utilize the URLs {3} through {n-1} again as shift indicators followed by the URL {1} to indicate the the preceding URL was a shift and not a word. This sequence is then followed by another URLfrom {3} to {n-1}to indicate the word desired. So, some words in the word dictionary (like "about" shown in Fig. 2) are mapped to a single URL and hence it takes only one URL request to convey such words. Other words of the word dictionary are mapped to multiple URLs (like "research in Fig.3). These words are conveyed by requesting the shift URL and a Wait URL followed by the word URL. The wait URL is used to indicate that for this particular word there exists multiple URLs mapped to it. Hence the receiver program must examine the URL following each initial URL to determine if it is a single or sequence of URLs. If a URL is followed immediately by a Wait URL then it is a mutliple URL or sequence.. In the example presented, "about" takes only one URL request which is URL number {3} while "research" requires three URL requests which are URL number {12} followed by the URL number {1}, which is the Wait URL (to indicate there is more than one URL mapped to the word research), and URL number {48}. The URL {12} being followed by the URL {1} indicates that it is a multiple URL or sequence and that the next URL indicates the specific word

The Wait URL allows this scheme to represent (n-3) single URL encodings and (n-3)*(n-3) multiple URL encoded words. Therefore if we have n URLs in the webpage we can encode a maximum number of words of

$$.\{\text{max words}\} = (n-3) + (n-3)^2$$

Phase 3: User Input: The final phase at the sender's side would include accepting URL numbers (mapped to words) from the sender in order to invoke respective URL requests.

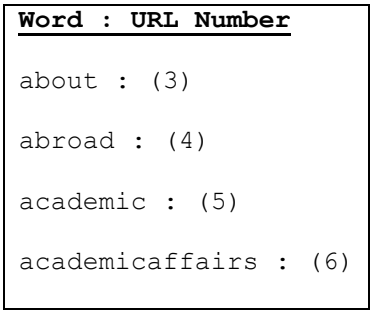

Figure 2. Single URL Mapping Sample

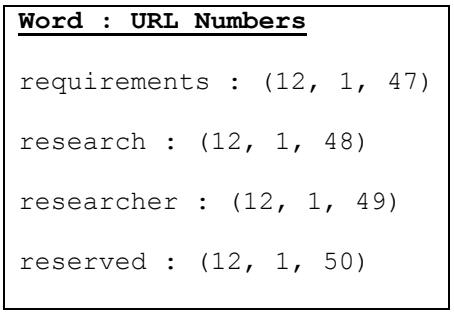

Figure 3. Multiple URL Mapping Sample

This way for every word of the covert message, mapped URL(s) are accessed. With reference to Fig. 2 and Fig. 3, for sending the covert message "about research", we must input the URLs {3} for about and {12, 48} for research. The same is illustrated in Fig. 4.

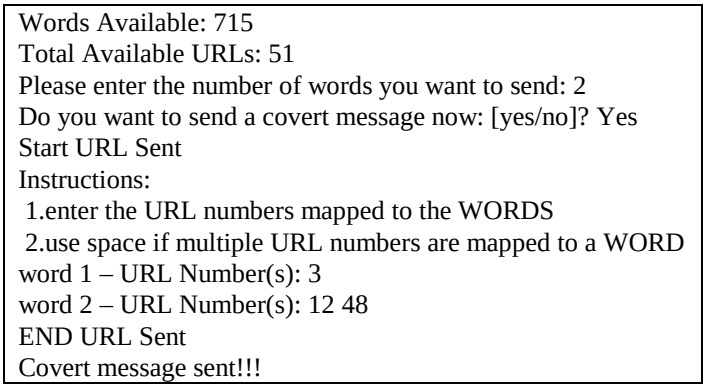

Figure 4. Send Covert Message

2) Receiver Side: The implementation on the receiver side is identical to the server side implementation except that the key-value pairs built for the dictionaries are reversed to facilitate decoding of covert message.

Phase 1: URL Dictionary: The first phase for both the sender and receiver are the same as it involves building a dictionary of URLs from the webpage that was decided prior to the communication. However, structurally the key, value pair of dictionary is reversed. In this implementation, the URL acts as the key while the numbers act as their corresponding values.

Phase 2: Word List & Mapping: After building the URL dictionary as mentioned above, the receiver builds a word list and then performs word-URL number mapping but in this case the URL number acts as key and its corresponding values are the words parsed from the webpage source.

Phase 3: Retrieval (PCAP/Apache Log): The final phase of the program on the receiver side is responsible for retrieving the message and displaying it to the receiver. This phase requires an additional input which is the packet capture file (.pcap) in our case. An apache access.log file is an alternative option. The reason for implementing using a packet capture is to imply that such a mechanism can be incorporated for other scenarios where a man-in-the-middle receiver, which is a system between the web client and web server, is the covert message receiver and captures the HTTP requests to decode the covert message. On receiving the packet capture file and sending entity's IP address as input, the receiver program retrieves the URLs requested by the sender from the packet capture and utilizes these URLs to decode the message sent as shown in Fig. 5.

```
H:\>python read.py

Enter your web page: http://www.rit.edu

Enter PCAP path: c:\captures\ritcovert.pcap

Enter Sender's IP: 192.168.0.6

Covert Message: about research
```

Figure 5. Receive Covert Message

5 Covert Channel Characteristics

5.1 Type

The proposed covert channel falls under the category of covert storage channel. This is because the sender's URL requests are captured in pcap (packet capture) file or the web server log and the receiver reads the packet capture or web server log to decode the message obtained.

5.2 Imperceptibility

It is very difficult for a regular observer or a capable administrator to detect the proposed covert channel as existing content of the webpage source and legitimate URL requests are utilized for its implementation and no alterations are made to the source code.

5.3 Robustness

The robustness of this channel is high considering the channel's ability of being persistent in the presence of firewalls, proxy servers, routers, etc. The message transfer is carried out by accessing URLs which are basically mapped to words using the encoding scheme. Since these URLs are legitimate HTTP requests they are not deterred by devices mentioned above.

5.4 Throughput

With reference to the word-URL mapping example described in the previous section, sending each word of the message takes either one or three URL requests. Every word either takes one or three URL requests as some words are mapped to a single URL and would hence require only one URL request while other words mapped to two URLs are implemented by requesting first URL followed by Wait URL and then the second URL. Considering this, a message having 10 words would take 10-30 URL requests excluding the Start URL and the End URL. This clearly indicates that the proposed covert channel demonstrates much higher bandwidth unlike the existing HTML based covert channels. Table II presents the number of URLs and words parsed from some pages at the time of writing this paper.

TABLE II
SOME WORD-URLSTATISTICS

Webpage	Words Available	URLs Available
www.rit.edu	730	51
www.cnn.com	1753	119
www.premierleague.com	931	174

5.5 Prevention

The imperceptibility factor plays a very crucial role in this covert channel. Though it is quite difficult to detect this covert channel, on identifying the existence of this covert channel, the channel can be disrupted by injecting HTTP requests for URLs existing in the URL dictionary with source IP address spoofed to that of the covert message sender.

6 Future Work

Though the proposed covert channel fares high on various characteristics, certain modifications and implementations can further improve the effectiveness of this channel. The message (word) dictionary part of the encoding/decoding mechanism can be improved by implementing an algorithm that would build the message dictionary from the webpage source on the basis of word occurrence frequency in the English language. This approach will help in ignoring words that are less likely to be used. Also, the implementation must be improved to ignore redirection of a URL to another URL that already exists in the URL dictionary as it might result in the transfer of incorrect messages. The widely varying sequence of URL requests would not be suspicious as it might look like a user browsing through different pages of a website however it might result in suspicion when these widely varying sequence of URL requests are made in a very short span of time by the same user. To prevent any suspicion or identification of the existence of this channel as a result of multiple URL requests from the same source in a short duration of time, the code can be modified to implement random wait times between multiple URL requests.

7 Conclusion

A communication channel, used to transfer secret message within the same system or across a network, where there is seemingly no communication taking place can be termed as covert channels. A new covert channel based on webpage source was exhibited in this paper. The covert channel presented, used words and URLs available in a webpage source to enable unidirectional covert communication from a web client to a web server. This covert channel does not modify the webpage source while demonstrating high imperceptibility and channel capacity.

8 References

[1] B.W. Lampson, "A note on the confinement problem,"*Commun. ACM*, vol. 16, no. 10, pp. 613–615, Oct. 1973. [Online]. Available: http://doi.acm.org/10.1145/362375.362389

[2] V. D. Gligor, "A guide to understanding covert channel analysis of trusted systems," *ser. Rainbow Series. National Computer Security Center*, Nov. 1993. [Online]. Available: http://www.fas.org/irp/nsa/rainbow/tg030.htm

[3] R. C. Newman, "Covert computer and network communications," in *Proceedings of the 4th annual conference on Information security curriculum development*, ser. InfoSecCD '07. New York, NY, USA: ACM, 2007, pp. 12:1–12:8. [Online]. Available: http://doi.acm.org/10.1145/1409908.1409922

[4] D. Johnson, B. Yuan, and P. Lutz, "Behavior-Based covert channel in cyberspace," in *4th International ISKE Conference on Intelligent Systems and Knowledge Engineering*, 2009, pp. 311–318.

[5] X.-G. Sui and H. Luo, "A new steganography method based on hypertext," in *Radio Science Conference, 2004. Proceedings. 2004 Asia-Pacific*, Aug. 2004, pp. 181 – 184.

[6] H. Huang, X. Sun, Z. Li, and G. Sun, "Detection of hidden information in webpage," in *Fuzzy Systems and Knowledge Discovery, 2007. FSKD 2007.Fourth International Conference on*, vol. 4, Aug. 2007, pp. 317 – 321.

[7] D. Shen and H. Zhao, "A novel scheme of webpage information hiding based on attributes," in *Information Theory and Information Security (ICITIS), 2010 IEEE International Conference on*, Dec. 2010, pp. 1147 –1150.

[8] Y. Yang and Y. Yang, "An efficient webpage information hiding method based on tag attributes," in *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, vol. 3, Aug. 2010, pp. 1181 –1184.

[9] W. Huba, B. Yuan, D. Johnson, and P. Lutz, "A HTTP cookie covert channel," in *Proceedings of the 4th international conference on Security of Information and Networks*, ser. SIN '11. New York, NY, USA: ACM, 2011, pp. 133–136. [Online]. Available: http://doi.acm.org/10.1145/2070425.2070447

[10] E. Brown, D. Johnson, B. Yuan, and P. Lutz, "Covert channels in the HTTP network protocol: Channel characterization and detecting Man-in-the-Middle attacks," *The journal of Information Warfare*, vol. 9, no. 3, Dec. 2010.