

A Taxonomy of Hyperlink Hiding Techniques

Guang-Gang Geng¹, Xiu-Tao Yang², Wei Wang¹, and Chi-Jie Meng¹

¹ China Internet Network Information Center, Computer Network Information Center, Chinese Academy of Sciences, Beijing, China, 100180
`{gengguanggang,wangwei,mengchijie}@cnnic.cn`

² Beijing Institute of Electronic System Engineering, Beijing, China, 100854
`xiutaoyang_temp@163.com`

Abstract. Hidden links are designed solely for search engines rather than visitors. To get high search engine rankings, link hiding techniques are usually used for the profitability of underground economies, such as illicit game servers, false medical services, illegal gambling, and less attractive high-profit industry. This paper investigates hyperlink hiding techniques on the Web, and gives a detailed taxonomy. We believe the taxonomy can help develop appropriate countermeasures. Statistical experimental results on real Web data indicate that link hiding techniques are very prevalent. We also tried to explore the attitude of Google towards link hiding spam by analyzing the PageRank values of relative links. The results show that more should be done to punish the hidden link spam.

Keywords: Web spam, link hiding, hidden spam, spam detection

1 Introduction

Most Web surfers depend on search engines to locate information on the Web. Link analysis algorithms [1], such as PageRank [2] and HITS [3], are usually used for Search engines ranking. Link analysis algorithms assume that every link represents a vote of support, in the sense that if there is a link from page x to page y and these two pages are authored by different people, then the author of page x is recommending page y . In particular, PageRank is the basis of Google's search technology [4].

Web spammers try to mislead search engines to make a high rank in search results [5]. In this context, hyperlink hiding techniques are often used to deceive search engines. Spammers hope that many small endorsements from these pages with hidden links result in a sizable PageRank for the target page. Several questions naturally arise: what link hiding techniques are the spammers using; and, how prevalent are hidden spam links on the Web? This paper attempts to answer those questions.

The rest of sections are organized as follows. Section 2 presents a literature review. Section 3 gives a comprehensive taxonomy of current hidden link spam techniques. Section 4 describes the experimental analysis on 5,583,451 Chinese Web sites. At last, section 5 draws the conclusion.

2 Related Work

Hidden links are designed to increase link popularity, which are invisible for visitors [6]. Google considers hyperlinks hidden by small characters as deception [7]. Gyongyi et al. point out that hidden links are often used in honey pot to boost the ranking of the spam pages [8]. They further present a comprehensive taxonomy of current spamming techniques and survey content hiding techniques, where spam links hidden by avoiding anchor texts or tiny anchor images are mentioned [5]. Link-hiding related features are not paid more attention to in statistical Web spam detection studies [9][10][11].

To the best of our knowledge, there is no previously published literature that directly studied how prevalent, successful, or varied hidden link spam techniques are on the Web. This paper attempts to study hidden link spam in detail. It is hoped that the findings can help in developing appropriate countermeasures.

3 Hyperlink Hiding Techniques

There are many different ways to hide links from visitors while leaving it perfectly viewable to search engines. In this section, we will examine current hyperlink hiding techniques used by spammers and attempt to categorize them based on their features. Just as the work on JavaScript redirection spam [12], we present short examples to show the hiding techniques really used by spammers. Simple techniques are presented first and are followed by more advanced ones.

3.1 A: Making Anchor Text Font Color the Same as Background Color

The simplest and oldest method that spammers use to create hidden links is to make the font of anchor text the same color as the background. Here is one example.

```
<span style="background:white;" >
  <a href="target.html" style="color:white"> invisible anchor text </a>
</span>
```

In this example, the color scheme is defined in the HTML document. Color schemes can also be defined in an attached cascading style sheet file (CSS). Sometimes, spammers also consider background images. They set the image color to be the same as the font color, which is relatively harder to detect.

3.2 B: Making Anchor Text Font Color Almost Match Background Color or Background Image

Instead of setting the font color to entirely match the background color, some spammers and web masters set their font colors to almost match the background color. The idea behind this method is that they believe that they are thwarting the search engines' software detection systems by slightly changing the color of the text.

```
<div style="background-color:white;" >
  <a href="target.html" style="color:#feffee"> text color similar to white </a>
</div>
```

3.3 C: Setting Tiny Anchor Text or Placing the hyperlinks in a Tiny Block

Making tiny anchor text is another hyperlink hiding method. This way, the hyperlink can be set small enough, such as 1 pixel high, even 0 pixel. Here's a simple example of that.

```
<a href="target.html" style="font-size:0px"> tiny text </a>
```

In HTML, the div element is often used for generic organizational or stylistic applications. Spammers can also use div to set the link size. The following is another example.

```
<div style="font-size:0px;"> <a href="target.html">invisible text</a> </div>
```

Perhaps the most common use of div element is to carry class or id attributes in conjunction with CSS to apply layout, typographic, color, and other presentation attributes to parts of the content. In the previous example, the *font-size:0px* can also be defined in a CSS file. Besides, div block size can be set via width and height attributes. For example, `<div style="width:1px;height:1px;">`, where the div size is 1 pixel.

Another example of hiding a hyperlink via tiny scrolling block is presented below.

```
<marquee scrollAmount=1 width=1 height=1>
  <a href="target.html"> text in a tiny scrolling block </a>
</marquee>
```

In this example, *target.html* is put in a scrolling block with area 1 × 1 pixel, which is invisible to Web users.

3.4 D: Disguising Anchor Text as Plain Text

Sometimes, spammers insert hyperlinks into a paragraph, where the anchor text looks like plain text. Here's a paragraph of text on a site:

The SEO company follows strict rules to insure the clients website reach the top of search engines and stay there.

A user wouldn't see any hyperlinks, even if they moused over every word in the paragraph. But if you happened to click on just the right word, you'd get whisked away to a SEO site. Actually, there is a hidden link under the anchor text "SEO company". If you view the source of the page, here's what you'll see:

```
The <a href="http://www.seomarketleaders.com" onMouseOver=
"window.status='';return true;" style="cursor:text;color:black;
text-decoration:none;"> SEO company</a> follows strict
rules to insure the clients website reach the top of search engines and stay there.
```

3.5 E: Placing Hyperlinks in High-Speed Scrolling Blocks

The `<marquee>` tag is a non-standard HTML element which causes text to scroll up, down, left or right automatically [13]. Although the W3C advises against its use in HTML documents, it's still widely used. `SCROLLAMOUNT` attribute sets the speed of the scrolling. A bigger value for `SCROLLAMOUNT` makes the marquee scroll faster. If the `SCROLLAMOUNT` value is big enough, the scrolling block will be invisible to the naked eye. Here is a simple example.

```
<marquee height=1 width=8 scrollamount=3000>
  <a href="target.html"> text in a high-speed scrolling block </a>
</marquee>
```

The default *scrollamount* value is 6. The value in the example is 3000, which is too fast to see.

Similar effects can also be achieved through the use of JavaScript or HTML `<blink>` element [13] [14].

3.6 F: Putting Links outside the Screen

Using cascading style sheets, you have the option to absolutely or relatively position any division. Using absolute position, you can simply position the text you wish to hide any number of pixels off the screen to the left of the window. Here are some example codes:

```
.hiddenclass { position : absolute;left : -977px; }
```

If you put that in your style sheet and then assign the class "hiddenclass" to your div, then the div will display 977 pixels to the left of the visible screen - i.e., it will not appear on the screen. Here is a example:

```
<div id="hiddenclass"> <a href="target.html"> invisible anchor text </a> </div>
```

The absolute position can also be set in the div directly as follows:

```
<div style="left: -977px; position: absolute; top: -977px">
  <a href="target.html"> invisible anchor text </a>
</div>
```

In the example above, *left* : -977 may be written in more complex formats, such as *left* : *expression*(23 - 1000).

In addition to the methods described above, users can use CSS text-indent property or margin-left property to put hyperlinks outside the screen. A example is presented below.

```
<div style="text-indent:-999px;"><a href="target.html">hidden text</a> </div>
```

3.7 G: Using Visibility:Hidden or Display:None Style Commands

An alternative to the method above is to simply use the built in features of style sheets to hide hyperlinks:

```
.hiddenclass { visibility : hidden;}
```

Again, if you put that into a style sheet and then assign the class “hidden-class” to your div, the hyperlinks in the div block will not appear in the browser window.

3.8 H: Hiding Hyperlinks via JavaScript

JavaScript is an open source programming language commonly implemented as part of a web browser in order to create enhanced user interfaces and dynamic websites [15]. Google claims that search engines have difficulty accessing JavaScript [7]. In 2011, labnol.org reported that Google indexes JavaScript based Facebook comments, but there is no clear report that Google parsers JavaScript codes on the whole Web. This fact encourages spammers to hide hyperlinks by the aid of JavaScript. Here is a simple example:

```
<script language="JavaScript" type="text/javascript">
  document.write( "<div style='visibility:hidden'>" );
</script>
<a href="target.html">keywords</a>
<script language="JavaScript" type="text/javascript">
  document.write( "</div>" );
</script>
```

The example is easy to understand, which is a packaging of the method described in section 3.7. In the above codes, <div> and </div> tags are embed in JavaScript codes separately, which may not be indexed by search engines. However, the hyperlink *target.html* is displayed in html codes, which is more likely to indexed by search engines. In a similar manner, almost all the link hiding techniques described in this section can be further disguised with JavaScript. Next, let's look into a more complex example.

```

<div id="ql1000">
  <a href="target.html" title="keyword">
    target keyword
  </a>
</div>
<script language="JavaScript">
var _xa= [
  "\x64\x69\x73\x70\x6C\x61\x79", "\x6E\x6F\x6E\x65",
  "\x71\x6c\x31\x30\x30\x30", "\x73\x74\x79\x6C\x65",
  "\x67\x65\x74\x45\x6C\x65\x6D\x65\x6E\x74\x42\x79\x49\x64"];
document[_xa[4]](_xa[2]][_xa[3]][_xa[0]]=_xa[1];
</script>

```

The above JavaScript codes are designed in rather vague terms. The elements of array `_xa` are written with ASCII characters. The last line of the above JavaScript codes is `document['getElementById']('ql1000')['style']['display']='none'`, which makes all the content, including hyperlinks, in the div named `ql1000` invisible. In order to avoid presenting the whole style assignment directly, script can build up the style assignment via string concatenation. One very straight forward example is presented below.

```

<script type="text/javascript">
  document.getElementById("q" + "l" + "1000").style.display="n" + "o" + "ne";
</script>

```

What is worse, JavaScript as a programming language, has many functions and operators, which throw off a human readers. The following codes show the flexibility of JavaScript.

```

<script language="javascript">function HexTostring(s){
  var r="";
  for(var i=0;i<s.length;i+=2){
    var sxx=parseInt(s.substring(i,i+2),16);
    r+=String.fromCharCode(sxx);}
  return r;}
eval(HexTostring(("646f63756d656e742e676574456c656d65
6e74427949642822716c3130303022292e7374796c652e6469
73706c6179203d20226e6f6e6522")));
</script>

```

These codes are essentially equivalent to the previous example, yet look completely different.

3.9 I: Hiding Hyperlinks via Cloaking or Redirection Techniques

Cloaking is a Web spam technique in which the page presented to the search engine spider is different from that presented to the user's browser [16]. Some

spammers hide target hyperlinks using cloaking technique. Similarly, spammers also use redirection techniques to hide targeting hyperlinks. Among the redirection spam techniques, JavaScript based redirection is the most notorious and difficult to catch [12]. Wu et al. [16] and Chellapilla et al. [12] have conducted comprehensive studies of cloaking and redirection techniques respectively, so the techniques will not be repeated here. However, it's important to point out that we do not consider the redirected target URL, but the hyperlinks in the redirection page as hidden links. For example, *A* redirects to *B*, and *C* is a hyperlink in page *A*. In this paper, *C* is a hidden link, but *B* is not seen as a hidden link.

3.10 J: Hiding Hyperlinks in Pull-Down Menu

Pull-down menu is also called a drop-down menu, which is a menu of commands or options that appears when you select an item with a mouse. A drop down menu can make it easier to display a large list of choices - since only one choice is displayed initially, the remaining choices can be displayed when the user activates the dropdown. Some spammers insert the target hyperlinks into a long pull-down list, which are hard to find.

3.11 K: Inserting Links into Long Title or Meta Tags

Generally, web browsers show the preceding part of a long title. Thus, some spammers insert urls into long title. Similarly, meta tags provide structured meta data about a Web page and they are used for search engines. Although they have been the targets of spammers for a long time and search engines consider these data less and less, there are pages still using them.

3.12 L: Hiding Div “Below” the Visible Layer

Another sneaky way to hide a hyperlink from Web users while keeping it available to the search engines is to put the hyperlinks in a layer that is “behind” the visible layer. The CSS z-index property specifies the stack order of an element, which is supported in all major browsers. An element with a greater stack order is always in front of an element with a lower stack order. One example hiding hyperlinks via z-index is presented below.

```
<div id="front" style="position:absolute; z-index:1">
  
</div>
<div id="back" style="position:absolute; z-index:-1">
  <a href="target.html" target="_blank"> target keyword</a>
</div>
```

The codes show that the second div has a negative stack order, which determines the *target.html* is behind the *image.gif*. Besides z-index, “overflow:hidden” can also hide the hyperlinks below the visible layer. Here is a simple example.

```
<style type="text/css">
#spam{width:99px;height:20px;overflow:hidden;position:absolute;}
#spam a{display:block;line-height:20px;text-decoration:none;}
</style>
<div id="spam">
  <a href="/">&#160;</a>
  <a href="target.html" title="keywords">
    target keyword
  </a>
</div>
```

In the example above, *target.html* is covered by a non-breaking space.

4 Prevalence of Link Hiding Techniques

Link-hiding can be considered an adversarial problem. As commercial search engines develop algorithms to detect and discard certain types of hidden links, new techniques for hiding links will be developed. In last section, we examined current hyperlink hiding techniques used by spammers and categorized them based on their features. In this section, we study the prevalence of hidden spam links, and how prevalence of the variety of techniques described in Section 3.

We carried out the analysis on 5,765,357 Chinese homepages (*http://www. + domain name*) in Sep. 2012, including .com, .net and .cn domain names. To detect the Web pages with hiding links, we first train a cost sensitive naive bayes classifier on 103 pages with hidden links and 271 normal pages. The cost sensitive model ensures a high recall of pages with hidden links. Then, we filtered the 5,765,357 pages with the trained model. The detection results contain quite a few false alarms, but it’s enough for us to analyze the prevalence of hidden links. By random sampling from the suspicious set and carrying out manual verification, we approximately determined the number of pages with hidden links. Table 1 tabulates the statistics in detail.

Table 1. Percentage occurrence of hidden link spam among Chinese Web pages

URL Type	Count / Total	Percentage
.com/.net/.cn	81775/5765357 = 1/70.5	1.42%

It is noticed that a number of Chinese pages use hyperlink hiding techniques. To analyze the prevalence of the variety of techniques described in Section 3, we randomly sampled 4727 pages with hidden links from .com/.net/.cn set. Each sampled hidden link spam page was manually analyzed. All the 4727 samples were labeled with the types of techniques they used. Besides, all the hidden links are extracted for further analysis. In total, 16767 unique target hyperlinks are hidden in the 4727 pages.

Table 2 describes the prevalence of hidden link techniques in detail. The table shows that the 4727 pages contain 16767 unique hidden links. F, G and H are the most popular link hiding techniques, which account for 75.3% of that total. These three techniques can be easily used to hide multiple hyperlinks. It can be observed that some of the 4727 web pages contain more than one link hiding technique.

Table 2. Prevalence of different link hiding techniques

techniques	number(percentage)=>number of hidden links
A	102 (2.2%) => 661
B	51 (1.1%) => 493
C	137 (2.9%) => 561
D	322 (6.8%) => 357
E	136 (2.9%) => 1987
F	1157 (24.5%) => 68661
G	511 (10.8%) => 30192
H	1888 (39.9%) => 51570
I	86 (1.8%) => 2071
J	151 (3.2%) => 527
K	255 (5.4%) => 103
L	53 (1.1%) => 779
All	4849 (4849/4727=102.6%) => 157962(unique links: 16767)

Are the 16767 target pages punished by the search engines? We do not know the detailed ranking strategy of commercial search engines, but we can explore this problem from a side by analyzing the PageRank values of the target hyperlinks. Google provides a public interface, toolbarqueries.google.com, for querying the PageRank values. Table 3 shows the average PageRank values of target hidden links and randomly selected 39756 urls from DNS resolution logs.

Table 3 shows that the 16767 hidden links have an average PageRank value 1.34, which is higher than that of the randomly selected urls. To some extent, the result means that Google needs to establish a more effective punitive mechanism for the hidden links.

We further analyzed the high-frequency words in the anchor texts of the 16767 target hyperlinks. The top 20 high-frequency keywords and the corre-

Table 3. Comparison of average PageRank values

	hidden links	randomly selected urls
Number	16767	39756
Average PageRanks	1.340	1.137

sponding types are described in figure 1. The statistics show that gambling sites, personal game servers and medical services are the main types of the hidden links. Most of the sites belong to shady or illegal industries.

Keywords	Type	Keywords	Type
百家乐	Gambling	皇冠现金网	Gambling
全讯网	Gambling	时时彩	Gambling
博彩通	Gambling	癫痫病	Medical service
太阳城	Gambling	世博	Gambling
皇冠网	Gambling	武动乾坤	Illicit game server
传奇私服	Illicit game server	金宝博	Gambling
办证	Fake certificates	盈丰国际	Gambling
牛皮癣	Medical service	神印王座	Online game
澳门赌场	Gambling	网通传奇	Online game
网址之家	Navigation site	足球比分	Gambling

Fig. 1. The high-frequency words in the anchor texts of the target hyperlinks.

5 Conclusion and Future Work

In this paper we presented a variety of commonly used link hiding techniques, and organized them into a taxonomy. We analyzed the prevalence of common link hiding techniques on the web. Just as the previous work on Web spam [5] [12], we argue that such a structured discussion of the subject is important to raise the awareness of the research community. Given that most of the sites using link hiding techniques are shady or illegal industries, more should be done to punish the hidden link spam.

In the future, we should pay more attention to two things. The first is studying link hidden spam on a bigger data set, which includes multilingual samples. The second is developing a proper countermeasure to address the problem as

a whole, despite the variety of different link hiding techniques. One possible solution draws support from maturing optical character recognition techniques (OCR) [17]. The motivation is that as a computer vision technique, OCR can only read the visible content on the Web page like humans. The snapshot of a Web page can be easily taken via some softwares, such as wkhtmltopdf [18] and snapshotter [19]. All the visual text on the snapshot image can be recognized via OCR techniques as *textVector*. If an anchor text does not exist in the *textVector*, the corresponding hyperlink is identified as hidden link. And, of course, the relative position of anchor text should also be taken into account.

Acknowledgment

This paper is supported by grants National Natural Science Foundation of China (Nos. 61005029, 61103138 & 61375039). This paper is the result of many beneficial discussions with one of our collaborators at a major search engine company, who wishes to remain anonymous.

References

1. A. Ng, A. Zheng, and M. Jordan, “Stable algorithms for link analysis,” in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 258–266.
2. L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: bringing order to the web.” 1999.
3. J. Kleinberg, “Authoritative sources in a hyperlinked environment,” *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.
4. S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” *Computer networks and ISDN systems*, vol. 30, no. 1, pp. 107–117, 1998.
5. Z. Gyongyi and H. Garcia-Molina, “Web spam taxonomy,” in *First international workshop on adversarial information retrieval on the web (AIRWeb 2005)*, 2005.
6. Wikipedia, “Spamdexing — Wikipedia, the free encyclopedia,” 2013, [Online; accessed 17-January-2013]. [Online]. Available: <http://en.wikipedia.org/wiki/Spamdexing>
7. Google, “Webmaster guidelines - webmaster tools help,” 2013, [Online; accessed 17-January-2013]. [Online]. Available: <http://www.google.com/webmasters/guidelines.html>
8. Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, “Combating web spam with trustrank,” in *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*. VLDB Endowment, 2004, pp. 576–587.
9. M. Erdélyi, A. Garzó, and A. A. Benczúr, “Web spam classification: a few features worth more,” in *Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality*. ACM, 2011, pp. 27–34.
10. Y. Liu, F. Chen, W. Kong, H. Yu, M. Zhang, S. Ma, and L. Ru, “Identifying web spam with the wisdom of the crowds,” *ACM Transactions on the Web (TWEB)*, vol. 6, no. 1, p. 2, 2012.
11. N. Spirin and J. Han, “Survey on web spam detection: principles and algorithms,” *ACM SIGKDD Explorations Newsletter*, vol. 13, no. 2, pp. 50–64, 2012.

12. K. Chellapilla and A. Maykov, "A taxonomy of javascript redirection spam," in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*. ACM, 2007, pp. 81–88.
13. Wikipedia, "Marquee element — Wikipedia, the free encyclopedia," 2013, [Online; accessed 19-January-2013]. [Online]. Available: http://en.wikipedia.org/wiki/Marquee_element
14. "Blink element — Wikipedia, the free encyclopedia," 2013, [Online; accessed 20-January-2013]. [Online]. Available: http://en.wikipedia.org/wiki/Blink_element
15. D. Flanagan, *JavaScript: the definitive guide*. O'Reilly Media, Incorporated, 2006.
16. B. Wu and B. Davison, "Cloaking and redirection: A preliminary study," in *First International Workshop on Adversarial Information Retrieval on the Web (AIR-Web05)*, 2005.
17. S. Mori, H. Nishida, and H. Yamada, *Optical character recognition*. John Wiley & Sons, Inc., 1999.
18. "wkhtmltopdf," 2013, [Online; accessed 20-February-2013]. [Online]. Available: <http://code.google.com/p/wkhtmltopdf/>
19. "Snapshotter," 2013, [Online; accessed 20-February-2013]. [Online]. Available: <http://www.mewsoft.com/Products/Snapshotter.html>