The Dark Side of Google

T.Ly and M.Papadaki

Network Research Group, University of Plymouth, Plymouth, United Kingdom
e-mail: info@cscan.org

Abstract

Google is the most popular, powerful searching tool and is used by pretty much all the web community. However, it is so powerful that Google can easily be turned into a very useful hacking tool, if misused by ill-intentioned people: in fact, by deliberately searching for confidential and sensitive information that the search engine may have inadvertently picked up, Google clearly shows us its dark side and it will indeed be possible to locate and exploit several targets across the web thanks to Google.Thus, lots of sensitive data, such as passwords, credit card numbers or even social security numbers are readily accessible to hackers, who would simply make use of Google to find them. In that way, Google is nowadays considered as a real double-edged tool and some alternatives should quickly be implemented to counterattack the Google Hacking phenomenon.

In that way, our project work would be to explore and analyse the general threat posed by Google hacking. The outlines of our work would then include the investigation through different real cases of security intrusions that involved Google hacking and would also propose some ways of detecting and responding to these types of attacks. Therefore, the preventive solution that we suggested would be to set up a GHH (Google Hack Honeypot), which would allow security specialists to study the profile of attackers, in order to anticipate their next move.

Keywords

Google Hacking, Google Hacking DataBase, Google Hack Honeypot

1 Introduction

Nobody can deny today that the company Google nearly controls all the majority of the huge web market, especially concerning the web search engines. In fact, the search engine Google gives us results and queries, which are so relevant and precise that it is certainly the most used in the world. Furthermore, searching web pages with the help of keywords is not the only ability of Google: in fact, the web search engine also makes the inventory of all images of websites, videos and message groups (such as USENET), etc. Thus, Google is growing everyday and becoming an unavoidable tool for all web users.

However, ill-intentioned people, such as hackers, are capable to misuse the web search engine, in order to exploit its powerful search algorithms, such as PageRank. As the web cache of Google is huge (the web cache represents all the data of web

sites registered by Google), the web search engine Google can deliberately become a hacking tool by searching for confidential information, such as passwords, credit card numbers, social security numbers or even FOUO (**F**or **O**fficial **U**se **O**nly) documents.

This amazing phenomenon is called the **Google Hacking** and it grows pretty fast those days. It is indeed a recent term that refers to the *art of creating complex search engine queries in order to filter through large amounts of search results for particular information* (Bausch, Calishain and Dornfest, 2006). At a more simple level, the Google hacking techniques would simply allow pirates to use/misuse powerful tools of Google in order to find sensitive information.

As it were, the more the web search engine Google grows, the more Google hacking also grows… In fact, even some trickiest tools appeared on the web market, such as the active **Google Hacking Database (GHDB)**: it is indeed a reference database in the field, which makes an inventory of all new Google hacking techniques. The database currently contains 1468 entries, included in 14 categories (GHDB, 2007), such as advisories and vulnerabilities, files containing passwords or pages containing network or vulnerability data. When a new Google hacking technique is discovered, people could add it in the database and it is everyday updated.

In that way, as Google is not ready to stop soon, the Google Hacking will not be willing to give up as well... and there will always have much more Google attackers... such as the master in the field, Johnny Long (see figure 1).

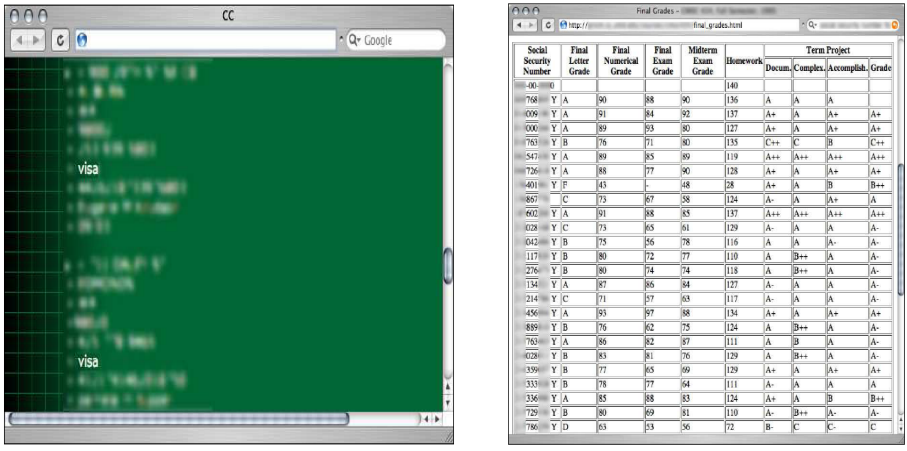

Figure 1: Credit card numbers & Social security numbers found by J. Long Source: Google Hacking for Penetration Testers (2004)

The main objective of this paper was to study the dark side of the search engine Google: the phenomenon as know as Google Hacking, mostly unknown to the general audience.

That is why the main purpose of this project will be to warn the general public, but rather people who build websites, i.e. the webmasters, of the threat posed by Google hackers. By the way, in order to better target the main aim, the project will be divided in 2 smaller objectives, which could also be compared to the main outlines of the project:

1. Reveal to all web users and especially to webmasters the real dangers of Google, which are generally unknown to the general public. The best way to achieve will be to explore the main techniques of Google Hacking and its latest trends.

2. Propose ways of detecting and responding to Google attackers: it will allow webmasters to better counterattack and also better understand Google's dark side.

2 Google Hack Honeypot, the reaction

2.1 The concept

The Google Hack Honeypot is not strictly speaking a real solution against Google hacking techniques. In fact, it is rather a 'reaction' to Google hackers (also called search engine hackers): the idea behind a Google Hack Honeypot (GHH) is that it places an invisible link onto your web site. Just like the case with a poorly constructed application, visitors to the web site will never see this link, but Google will. However, instead of providing access to confidential data, the link will conduct Google hackers to a PHP script that logs their activity.

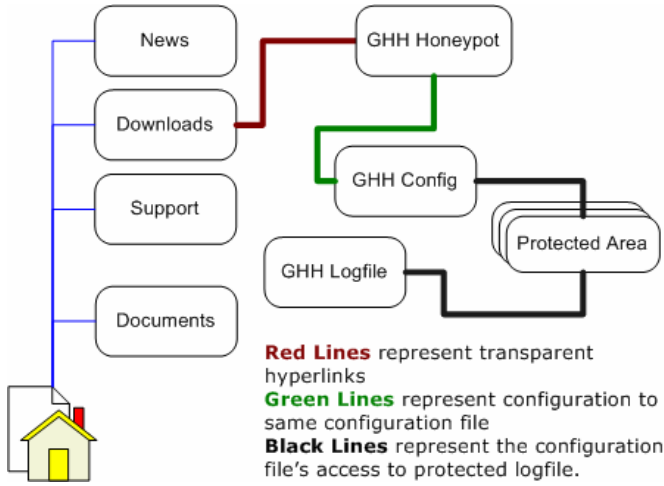

Figure 2: Google Hack Honeypot (GHH)
Source: http://ghh.sourceforge.net

In that way, security researchers and specialists would be able to draw an accurate profile of the attacks/attackers, in order to prevent and anticipate their next move.

At a simple level, a honeypot (hardware or software) will emulate vulnerabilities, or will deliberately contain some flaws and even some (false) confidential information: hackers will be attracted by this kind of information, and will fall in the trap: web administrators could also block hackers (IP address) and monitor how they launched their attacks (source of the attack), in order to prevent the next ones.

Therefore, GHH appears to be the perfect tool to better analyse and understand the Google Hacking phenomenon.

2.2 Experiments tested

Three different GHH experiments were launched through January to May 2007:

- Honeypot 1: GHDB Signature #935 (inurl:"install/install.php")
- Honeypot 2: GHDB Signature #1064 (filetype:sql ("passwd values" | "password values" | "pass values"))
- Honeypot 3: GHDB Signature #1758 "The statistics were last updated" "Daily"-microsoft.com

The honeypots were implemented in the same time and into a same hosted website. As it were, their set up was pretty easy to execute: there were indeed no specific need to install a hardware solution, nevertheless only few knowledge in PHP (the honeypots are coded in that web language) were necessary to manipulate them, but otherwise their implementation was quite simple to install. However, it is worth noting that the indexing into the Google search engine was a lot more challenging and it involved several steps; identification of the website, listing of all web pages with Sitemap (XML), implementation of the accurate META tags into the source code, waiting for a validation from a Google bot, which has to list all the webpages of the website, etc. In fact, in order to be seen by the hacker community, the opened vulnerabilities that the honeypots suggest and the website in itself as well have to clearly be visible on the web and particularly directly on the Google search engine. It is also important to that the website did not contain any confidential data (it was not a commercial website to be clear but rather a personal one, such as a blog). The purpose of potential attackers was then not financial. Perhaps, the website would have attracted more attacks, had it contained financial data.

The results were conclusive, as we caught all the same around 200 attacks the first month. However, the number of total attacks decreased month to month. The reason of this is probably that the 3 honeypots were hosted in the same website, so that the hackers did discover the traps. This suggests that the average life for a honeypot is about 1 to 2 months: after that ‘hackers’ begin to lose interest.

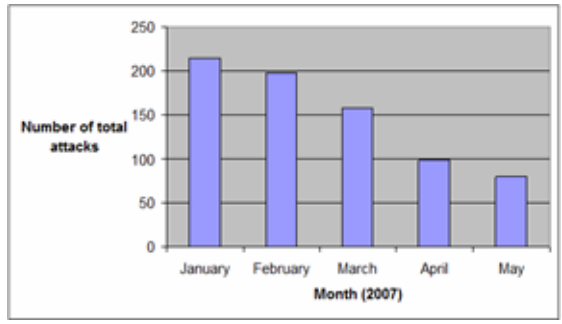

Figure 3: Number of total attacks

2.2.1 Honeypot 1: GHDB Signature #935

Collected data from the first honeypot reveal that the attackers were mainly script kiddies (inexperienced newbies in hacking). There is one thing to prove it: in fact, we remarked that the same IP consecutively appeared twice to four times, and even 5 times. When some script kiddies indeed fall in the honeypot, they stay in the same page, whereas they thought that they made a great exploit and are waiting for a new page with juicy information, but there is nothing for them, as they are in the honeypot! So, their reflex would be to hit again, by refreshing (actualizing) another time the webpage, and that is why we often caught the same IP in the results. Also, a large number of IPs belongs to some anonymous proxies, which is indicated by the fact that there was no specific response from whois queries (Whois, 2007). The attackers are then considered as anonymous. The IP addresses of those proxies are mainly from networks in Russia, Germany and China (see the graph), where lots of anonymous proxies' lists could even be rent. Moreover, as the website is hosted in Europe (from a French web host company) and then on Google France, the 'Other' IPs mostly did correspond to French IP addresses

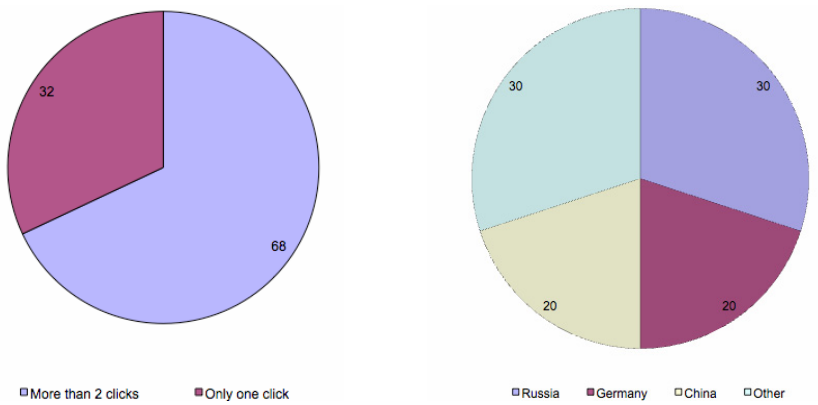

Figure 4: Script kiddies vs experienced hackers, Country of origin

2.2.2 Honeypot 2: GHDB Signature #1064

After a deep investigation, i.e. whois and traceroute (Whois, 2007)(Traceroute, 2007), most of the attacks come from a free tool called SiteDigger (the IPs are flagged with the SiteDigger signature, i.e. with a specific IP address). This software was created by Foundstone, a division of the famous antivirus company McAfee and was originally developed for helping security professionals to find (Google Hacking) web vulnerabilities (Foundstone, 2006). However, as it is the same issue for a lot of security tools, these are some double-edged weapons. In fact, SiteDigger also allows hackers to search for vulnerabilities, errors, configuration issues, proprietary information and interesting security nuggets on websites through Google. And icing on the cake, SiteDigger enables to directly look for vulnerabilities and signatures belonging to the unavoidable Google Hack DataBase (GHDB), which is a real incubator for Google hackers. In that way, the tool is always updated with the latest trickiest signatures of Google hacking.

Therefore, SiteDigger also allows hackers to automate their juicy information's searches thanks to Google (such as passwords) and this tool then appeared to be really dangerous. Nonetheless, the reaction of Google was, for once, quite effective: in fact, for activating SiteDigger, a Google API (Application Programming Interface) Key is requested, and it is the assault course to get the precious key: a large number of complex (and deliberated) registrations are required. Those were necessary to let Google to trace every query that SiteDigger will request (and by the way, the queries are limited to 1000 a day).

Furthermore, what it will be interesting to notice is that for the other honeypot, it was possible for hackers to hijack their identities by spoofing their IPs with some anonymous proxies. But for this specific case, all hints of identification through SiteDigger are logged and traced and it is not possible to be anonymous.

Nevertheless, many (good) black hat hackers asserted that it is very easy to create a tool such as SiteDigger: there is incidentally some unofficial tools which are distributed within the hacker community and that could easily be downloaded on the web.

As it were, despite the general Google's precautions, Site Digger is obviously used to look for unauthorised purposes. The evidence to support this claim is the fact that they searched your site for vulnerabilities, without your authorisation.

2.2.3 Honeypot 3: GHDB Signature #1758

It is really hard to draw a profile of the real attackers. Then what it would be interesting to study here is the kind of browser that the 'attackers' used. And results were pretty amazing: 40% of the users were using Internet Explorer (version 6 or 7), 35% (Mozilla) and 5% for the others (Opera, Netscape).

Mozilla is a browser that is said to be more secured (Zdnet, 2005), and people are more and more using it (Slate, 2004). However, what we could notice is that people did not update their version as we might do it, and that is pretty dangerous, as we know that old versions of browsers are subject to many attacks (Symantec Internet Security Threat Report, 2006).

Regarding now the country of origin of the attacks, they are coming by a majority from Europe (45%), US (20%), China (15%), Russia (15%) and others (5%). The reason why there are more attackers in Europe is because the website was hosted in France and will then be subject to more queries from European Google servers.

Basically, here are the other conclusions that we draw after analysis of the results:

- Many attacks appeared to come from scripts kiddies (inexperienced newbies in hacking). In fact, the Google hacking techniques through the GHDB for instance are likely very easy to use (such as making of use of Google itself).
- A large number of IPs belongs to some anonymous proxies and the attackers are then considered as anonymous. The IP addresses of those proxies are mainly from networks in Russia, Germany and China, where lots of anonymous proxies' lists could even be rent.
- Many IPs are coming from the strange same domain and networks. (e.g. x.x.x.1/20 with 10 IP addresses). This indicates that the IPs might belong to a potential botnet, as the time of the attack is very close to each other (even a simultaneous attack). The potential assumption is also that those particular IP addresses are probably all infected with a specific worm and then are hosted behind a specific ISP with DHCP (Dynamic Host Configuration Protocol) that keeps getting online/offline
- 40% of the users were using Internet Explorer (version 6 or 7), 35% (Mozilla) and 5% for the others (Opera, Netscape). We did noticed that the users did not update their version as we might do it, and that is pretty dangerous, as we know that old versions of browsers are subject to many attacks (Symantec Internet Security Threat Report, 2006).

To conclude this section regarding the Google Hacking experiments, what we could say is that the experiments were a great success: we caught many attacks and attackers, and it was pretty easy to draw conclusions (even if they were not as deep as we could expect). As it were, the honeypot is still an unknown technology, but it begins to be used more and more: in fact, its techniques are closely linked with the intrusion detection systems (Honeypots.net, 2007) and it is obvious that large companies will need it for their corporate website.

It is sure that those experiments were not perfectly raised at all: in fact, what it could be a potential improvement for better and more detailed analysis would be to set up one honeypot into one only hosted website: their shelf life (i.e. the honeypots were too easy to spot by hackers) would be surely longer and the collection of data would probably be more accurate.

Furthermore, what we noticed is that the average life for a honeypot is about 1 to 2 months: in fact, after catching attacks in the net, 'hackers' begin to understand that there noting really juicy in the website. In that way, they begin to give up and that is why we got less attacks during the last months.

Anyway, the honeypot technology is a good prospect: the next generation of honeypots would have to be more active and responsive. That is what the third generation of honeypots, as know as GenIII (honeywalls), recently appeared.

3 Conclusions and future work

This paper, which concluded 6 months of work on the topic, generally introduces the dark side of Google and its main concept based on some tricky Google Hacking techniques.

By the way, the study of the phenomenon was complete, as we get through different angles of attack. In fact, we addressed some deep theoretical points, by discussing the main techniques and also by reviewing some past incidents involving the Google hacking and we analysed with a practical viewpoint some (Google Hack Honeypot) experiments, in order to directly understand the purpose of the attackers.

As for a potential future work, there are plenty of choices: in fact, the Google Hacking phenomenon is not ready to stop for a good while. In fact, as the web search engine Google keeps growing, its dark side would keep increasing as well.

4 References

Dornfest, R., Bausch, P. and Calishain, T. (2006). "Google Hacks". Publisher: O'Reilly Media, U.S.A

GHDB (2007). "Google hacking DataBase", http://johnny.ihackstuff.com/ghdb.php [Accessed 30/08/2007]

Google Corporate History (2006). http://www.google.com/intl/en/corporate/security.html [Accessed 30/08/2007]

Google Hack Honeypot (2007). http://ghh.sourceforge.net/ [Accessed 30/08/2007]

Long, J. (2005). "Google Hacking for Penetration Testers". Publisher: Syngress Publishing, U.S.A

Symantec Internet Security Threat Report (2006). Trends for January 06 – June 06. Volume X, Published September 2006