Maciej Laskowski

USING GOOGLE SEARCH ENGINE TO GET UNAUTHORIZED ACCESS TO PRIVATE DATA

This article shows – by performing a number of test searches – how unauthorized users may gain access to private or proprietary data using popular Internet search engine. The simplest ways of data protection are also presented.

Keywords: *search engine; Google; confidential data; information protection.*

Мацей Ласковські

ВИКОРИСТАННЯ ПОШУКОВОГО ДВИГУНА GOOGLE ДЛЯ ОТРИМАННЯ НЕЛЕГАЛЬНОГО ДОСТУПУ ДО ОСОБИСТОЇ ІНФОРМАЦІЇ

У статті представлено кілька прикладів того, яким чином треті особи можуть отримати доступ до особистої або конфіденційної інформації, використовуючи найпопулярніший пошуковий двигун. Представлено також найпростіші способи захисту інформації від подібних посягань.

Ключові слова: *пошуковий двигун; Google; конфіденційні дані; захист інформації.*

Рис. 4. Літ. 9.

Мацей Ласковски

ИСПОЛЬЗОВАНИЕ ПОИСКОВОГО ДВИЖКА GOOGLE ДЛЯ ПОЛУЧЕНИЯ НЕЛЕГАЛЬНОГО ДОСТУПА К ЛИЧНОЙ ИНФОРМАЦИИ

В статье на нескольких примерах показано, каким образом третьи лица могут получить доступ к личной или конфиденциальной информации, используя наиболее популярный поисковый движок. Представлены наиболее простые способы защиты информации от подобных посягательств.

Ключевые слова: *поисковый движок; Google; конфиденциальные данные, защита информации.*

1. Introduction. Nowadays Google is considered to be the best, the fastest and the most reliable free search engine [1]. It attracts over 70% of searches, followed by Yahoo (almost 17% of searches) and Bing (almost 10%) [2]. In some countries Google domination is even more evident – in Poland, for example, Google has almost 95% of web search engine market share [3]. Most of the top search properties worldwide experienced significant growth in search query volume each year since 2005 [4].

Special applications (called Googlebots) scan the Web constantly, gathering information in order to provide better searching service. Basically, Google was designed just to allow even the most uninitiated web surfer to find what he/she was looking for. Despite many positive sides of its features (as images, newsgroups, news searches, including even language and document translation etc.) most of them offer far more nefarious possibilities to malicious Internet users (like hacker/cracker community, identity thieves etc.).

[1] Junior Assistant Professor, Institute of Computer Science, Lublin University of Technology, Poland.

In order to know how that could be possible, the methods of the Googlebots' work should be analyzed. It should be emphasized that majority of the techniques presented in this article can be successfully used with other search engines. The author has chosen this particular search engine due to two reasons: its leading role at the market and the widest working spectrum.

2. How do the Googlebots work? Googlebot is defined as "Google's web-crawling robot. It collects documents from the web to build a searchable index for the Google search engine" [5].

The creators of Google came across a very complicated problem – there is no easy way to index all webpages. At the time Google was developed, the majority of web search engines were based on the addresses added manually to their databases by webmasters. This narrowed down the range of the websites to be searched through.

One of the methods to avoid such narrowing was to follow conception of information being filtered by Internet users themselves – let's imagine such a situation: somebody puts on his website a link, directing to another one – this may mean that he considers the other one as interesting in some kind of way and (as the result) this page should be indexed in the database of a search engine. This is the main idea behind Google web search – the Googlebots crawl the network in two types of crawl – fresh crawl (run every day, searching for new data) and main crawl (which is being run about once a month) following each link they come across (except for pages disabled for crawling by some blocking techniques).

Basically this situation is completely correct and acceptable – but what if it means also configuration, password or private data files being accessible to everyone? An attacker can gain unauthorized access to private or proprietary data – all by using a popular search engine by only formulating a specified query.

3. Basic Google Search Techniques. Mainly to perform a quick search using Google, a user should enter a search phrase into the textfield on Google search main page.

But entering, for example, "john smith" will return not only pages referring to Mr. John Smith, but also pages containing both john and smith (e.g., pages about actors John Wayne and Will Smith). Using the plus sign (+) (with no space following, e.g.. +john +smith) forces Google to perform a search for an overly common word, while using the minus sign (-) excludes a term from a search. Searching for an exact phrase requires entering it between double quotes (e.g. "john smith"). A period (.) serves as a single-character wildcard, while the asterisk (*) can act as any word – but not as a multiple-character wildcard [3].

Google also accepts Boolean logic – AND operator is automatically added between each word in a query. But if OR is used – the results will show sites containing either one word or the other, while using NOT will result in pages containing all the queried words except the one (or the ones excluded).

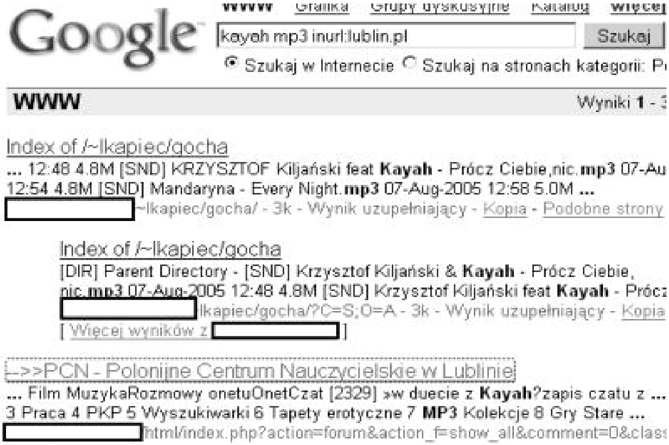
4. Advanced Search Techniques. A query can contain some predefined advanced operators, used with the following syntax: operator:search_term [6]. The most popular operators are [6]:

- site: – narrowing a query to a specific website or domain name;
- filetype: – narrowing a query to a particular type of files, e.g. pdf. It is important not to include a period (.) before the file extension;
- link: – search is performed within hyperlinks for a search term;

- cache: – displaying the page as it appeared during the last crawl (if possible);
- intitle: – query is performed within titles of documents;
- inurl: – search is limited only to documents' URL.

5. Finding private/proprietary data with Google. Having the knowledge of using Google's operators combined with basic search techniques every user can perform a specific query – either to gain access to personal/proprietary data or to find some system vulnerabilities. This technique is widely known as Google hacking.

Google hacking is most frequently used to find private or proprietary data. For example: a user in Lublin is searching for some music by Polish singer Kayah. He's especially interested in servers located in Lublin (as he assumes the download should be at least a little faster). Querying for kayah mp3 inurl:lublin.pl returns 3 matches. One is just a text-based website, but the other two are fully indexed directories of proprietary mp3s, unprotected in any way (see Fig.1 for details).



kayah mp3 inurl:lublin.pl

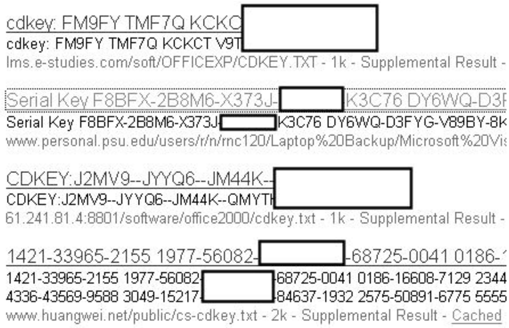
Index of /~lkapiec/gocha
... 12:48 4.8M [SND] KRZYSZTOF Kiljański feat **Kayah** - Prócz Ciebie,nic.**mp3** 07-Au
12:54 4.8M [SND] Mandaryna - Every Night.**mp3** 07-Aug-2005 12:58 5.0M ...
[redacted]~lkapiec/gocha/ - 3k - Wynik uzupełniający - Kopia - Podobne strony

Index of /~lkapiec/gocha
[DIR] Parent Directory - [SND] Krzysztof Kiljański & **Kayah** - Prócz Ciebie,
nic.**mp3** 07-Aug-2005 12:48 4.8M [SND] Krzysztof Kiljański feat **Kayah** - Prócz
[redacted]lkapiec/gocha/?C=S;O=A - 3k - Wynik uzupełniający - Kopia
[Więcej wyników z [redacted]]

-->>PCN - Polonijne Centrum Nauczycielskie w Lublinie
... Film MuzykaRozmowy onetuOnetCzat [2329] »w duecie z **Kayah**?zapis czatu z ...
3 Praca 4 PKP 5 Wyszukiwarki 6 Tapety erotyczne 7 **MP3** Kolekcje 8 Gry Stare ...
[redacted]html/index.php?action=forum&action_f=show_all&comment=0&class

Fig. 1. **An example of finding proprietary files using specific query (source: own work)**

Looking for unspecified cd-key using allinurl:cdkey.txt will result in a number of hits. Specifying a software name, e.g. by using allinurl:cdkey.txt office, will narrow down the results number. And even if the indexed websites were removed from the Internet, almost all of the data needed is shown in search result window, as it is shown in Fig. 2.



cdkey: FM9FY TMF7Q KCKC[redacted]
cdkey: FM9FY TMF7Q KCKCT V9T
lms.e-studies.com/soft/OFFICEXP/CDKEY.TXT - 1k - Supplemental Result -

Serial Key F8BFX-2B8M6-X373J-[redacted]K3C76 DY6WQ-D3F
Serial Key F8BFX-2B8M6-X373J[redacted]K3C76 DY6WQ-D3FYG-V89BY-8K
www.personal.psu.edu/users/r/n/rnc120/Laptop%20Backup/Microsoft%20Vis

CDKEY:J2MV9--JYYQ6--JM44K--[redacted]
CDKEY:J2MV9--JYYQ6--JM44K--QMYT
61.241.81.4:8801/software/office2000/cdkey.txt - 1k - Supplemental Result -

1421-33965-2155 1977-56082-[redacted]-68725-0041 0186-
1421-33965-2155 1977-56082[redacted]68725-0041 0186-16608-7129 2344
4336-43569-9588 3049-15217 84637-1932 2575-50891-6775 5555
www.huangwei.net/public/cs-cdkey.txt - 2k - Supplemental Result - Cached

Fig. 2. **Google's result window for allinurl:cd-key.txt search query (source: own work)**

Google hacking can be a much more serious matter than finding proprietary data on the Internet: some users block access to their websites by using .htpasswd files. This is a very good way to disabe unauthorized access, as these files are executed server-side, but under some circumstances (especially non-standard filename and wrong set of file permissions) they can be indexed and accessible via Google. Moreover, it should be noticed that Google itself introduced a special file type called htpasswd. This illustrates the scale of problem.

Quering for filetype:htpasswd htpasswd returns over 1500 hits, with logins and encrypted passwords (which in some cases can be decrypted [7]), e.g.

dave: 1eAziU9yTB.16

udel: f2JnN2GXS1CXI

Another method of gaining access to .htpasswd files is to perform a query for intitle:"index of" ".htpasswd" "htgroup" – intitle: "dist" – apache – htpasswd.c. This search returns with significantly less number of hits (around 60), often resulting in error 403 (Forbidden) – this method is less successful then previous one.

Searching for password.log files is also a popular method of gaining access to private data. Searching for filetype: log inurl: "password.log" results in over 600 possible hits. Typical password.log file contains: login, password (often in plaintext) and URL, where the user should log in, e.g.

name: = "test";

password: = "test";

URL: = "passdemo1.html";

An aggressor can easily use this information to gain unauthorized access to a website.

Google can also help attackers steal identity of another Internet user. Many people publish a number of documents online, which include some very private data (as Social Security Number, address, ID card number etc.). By using this data an attacker is able, for example, to apply for a credit card in some countries. All has to be done is to search for "phone * * *" "address *" "e-mail" intitle:"curriculum vitae" (almost 41000 hits), or more specified, "social security number" "phone * * *" "address *" "e-mail *" intitle:"curriculum vitae", which returns only about 150 hits. In just a couple of minutes an attacker can gain access to very private and confidential data.

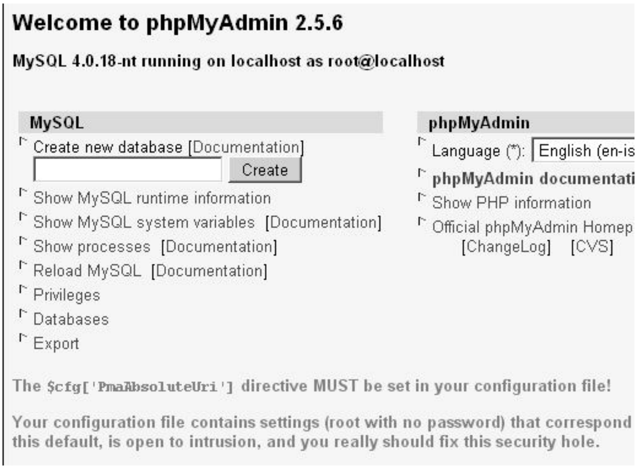
Google hacking can also be used for building spam lists, as some people export their address boxes as .csv files and put them online (e.g., for backup purposes). Querying e-mail address filetype:csv csv results in over 600 hits and thousands of e-mail addresses and often more personal information, like phone numbers and postal addresses.

Nowadays, especially in business, some documents are crucial and shall not be accessible to anyone except interested parties. By searching for "not for distribution" confidential, an attacker instantly may be given access to over 12 thousand of confidential documents, which shouldn't be distributed worldwide.

6. Finding system vulnerabilities with Google. Google hacking means an attacker is able to find system vulnerabilities.

Content Management Systems (CMS) are quite popular even among common Internet users. But some of them run their system without securing it properly, for example, leaving the configuration of MySQL database with root privileges without a password. Searching for intitle:phpMyAdmin "Welcome to phpMyAdmin ***" "running on * as root@*" returns almost 7000 hits, most of them being still active.

Fig. 3 and 4 show the unauthorized modifications of database structure.



Welcome to phpMyAdmin 2.5.6

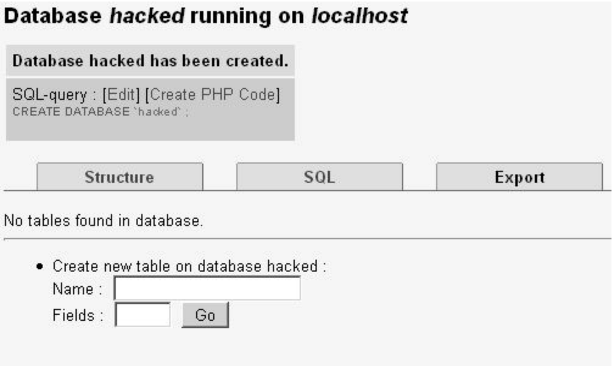
MySQL 4.0.18-nt running on localhost as root@localhost

MySQL
Create new database [Documentation]
Create
Show MySQL runtime information
Show MySQL system variables [Documentation]
Show processes [Documentation]
Reload MySQL [Documentation]
Privileges
Databases
Export
phpMyAdmin
Language (*): English (en-is
phpMyAdmin documentati
Show PHP information
Official phpMyAdmin Homep
[ChangeLog] [CVS]

The $cfg['PmaAbsoluteUri'] directive MUST be set in your configuration file!

Your configuration file contains settings (root with no password) that correspond this default, is open to intrusion, and you really should fix this security hole.

Fig. 3. **Welcome screen of unprotected phpMyAdmin 2.5.6 (source: own work)**



Database *hacked* running on *localhost*

Database hacked has been created.

SQL-query : [Edit] [Create PHP Code]
CREATE DATABASE `hacked` ;

No tables found in database.

- Create new table on database hacked :

Name :

Fields :

Fig. 4. **Modifying the database structure by adding database 'hacked' (source: own work)**

7. Basic methods of protecting data from unauthorized access via web search engines. First of all, users should not put sensitive data on the web, even temporally, especially without securing it with .htaccess file. Moreover, web server administrator should turn off listing the files in the directories in case the index file (index.html, index.php, etc.) is not present. If it is not possible (e.g., in case of having unprivileged user account on a server), webmaster should place a blank index file in each directory (even those seeming insignificant) or a .htaccess file, e.g., with rule redirecting a user to the main page of a website.

The second basic rule is to use tools provided by web search engines themselves for disabling indexation of a website or some part of it. Both robots.txt file and <a>

tag are considered as a respected standard and the crawling software of major search engines do recognize robots.txt file and its contents. Of course, even robots.txt file can be used by attackers to gain knowledge about the structure and protected directories and files, but this issue seems to have no solution at the moment, as this file has to be accessible by everyone.

Scanning a website for finding vulnerabilities (for example, the ones described in this article) is also quite a useful method. Moreover, the source code of website should be cleared off any unnecessary links. This should result not only in fixing some of the problems described in this paper, but should also speed up the website indexing by searchbots [5].

The last, and probably the most aggressive method is to remove website from web search engine index. But in many cases the data removed from the Web may be still (at least partially) accessed by web search engine cache.

8. Summary. Unauthorized access to private data put online is nowadays considered as a moderate danger for an experienced webmaster. But as it is proven by examples above, there are still many Internet users not aware of possible dangers. The problem is especially important in cases of commercial data, which should be confidential.

The value of data gathered by malicious web search queries varies, but an attacker may be able (in some cases) to gain access either to private data (like passwords or personal information) or to confidential documents at practically no time. Securing private data put online is especially important, as in many cases users are not aware that e.g. placing online a CV with their SSN can be more dangerous in case of identity theft attempt than having a malicious software installed [9].

It should be noticed, that with the popularization of this subject, the awareness of webmasters and programmers grows constantly, while number of methods of possible attack decreases. Moreover, the search engines themselves try to reduce the number of malicious queries by blocking some of query possibilities or removing some of the indexed files or filetypes from their databases.

Bibliography

[1] *Piotrowski M.* Niebezpieczne Google – wyszukiwanie poufnych informacji, hakin9 3/2005 (in Polish).

[2] http://www.seoconsultants.com/search-engines/ (accessed 29.05.2011).

[3] http://www.firstlevel.pl/udzial-rynku-wyszukiwarek-w-polsce/, (in Polish, accessed 30.08.2011).

[4] http://www.comscore.com/Press_Events/Press_Releases/2009/8/Global_Search_Market_Draws_ More_than_100_ Billion_Searches_per_Month (accessed 30.05.2011).

[5] http://www.google.com/webmasters/bot.html (accessed 30.05.2011).

[6] *Long J.* Google hacking mini-guide, http://www.informit.com/articles/article.asp?p=170880&rl=1 (accessed 30.05.2011).

[7] *Liu H.., Pallickara S., Fox G.* Performance of Web Services Security, http://grids.ucs.indiana.edu/ptliupages/publications/ WSSPerf.pdf (accessed 31.05.2010).

[9] http://www.idtheftcenter.org/artman2/publish/c_tips/Fact_Sheet_117_IDENTITY_THEFT_AND_THE_DECEASED_-_PREVENTION_AND_VICTIM_TIPS.shtml (accessed 31.05.2010).

Стаття надійшла до редакції 10.01.2012.