

Googling Forensics

Benjamin Turnbull
University of South Australia
benjamin.turnbull@unisa.edu.au

Detective Sergeant Barry Blundell
South Australia Police (Electronic Crime Division)

Dr Jill Slay
University of South Australia

Abstract

This paper discusses the emerging trend of Personal Desktop Searching utilities on desktop PCs, and how the information cached and stored with these systems can be retrieved and analysed, even after the original document has been removed. Focusing on the free Google Desktop Search program, this paper first analyses how the program operates, the processes involved, files created and altered, and methods on retrieving this data without corrupting the contents. The limitations of extracting data from Google Desktop Search have also been discussed, along with some future work in the area.

Keywords

Forensic computing, computer forensics, Google, desktop search

INTRODUCTION

As computer usage continues to become more ubiquitous, the data created, stored and edited by the average user has grown in variety, complexity and quantity. Email, word processing, basic text, accounting, video and audio are just a small number of file types that the average computer user may utilise. Whilst searching for files is a feature found in the majority of Operating Systems, the complexity and range of data on the modern PC has left them limited in their usage, awkwardly slow and unable to navigate within documents, leaving them useless unless searching by file names.

This technology gap has recently become a contested area between several companies with Internet search engines, as well as a number of small start-up enterprises. The attractiveness of this new market and being in a position to merge desktop and Internet searching – can in effect ensure more clients for a particular online search site.

The uptake in these programs may have benefits within the field of Forensic Computing. In essence, these indexed files may take much of the drudgery away from searching entire hard disks for keywords, when the majority of user data may already be indexed by one or more search utilities. Whilst it is expected that there will be limitations, any program that stores metadata may be of use within an investigation, as that is the primary purpose of these tools.

This paper aims to analyse one popular desktop search program, Google Desktop Search, discuss how it operates, if and where it stores data, and the limitations of its operation. All data has been collected on dedicated machines utilising no other software that may interfere, and where analysis software has been used, it has been chosen for its unobtrusive and passive nature. Google automatically updates its desktop search program via HTTP, so it is difficult to discuss versions of the program. All experiments were carried out between the 11th and 26th May, 2005.

GOOGLE DESKTOP SEARCH

Google Desktop Search was one of the first programs released onto the public market in mid 2004, and despite only recently leaving the Beta testing stage it represents one of the more popular desktop searching utilities. It is designed for use on a single-user Windows machine. Within a multi-user environment, should a user with administrative rights install and run Google Desktop Search, the program indexes and searched all users' files, regardless of their owner.

Google experienced negative publicity from a number of sources after the initial release of the product which was widely reported in the press, with many citing it as a potential security weakness (Spring, 2004; Posey,

2005). Google Desktop Search merely indexed all files that it is given access to, highlighting the security issues of multi-user systems and Windows reliance on administrative accounts rather than causing these issues. To many, this represents a failure in effective design if not security.

Google Desktop has also had other bugs discovered within it, resulting from a study conducted by Rice University, indicating that vulnerabilities existed in the integration of Google Desktop Search and the Google internet search engine (Nielson et al, 2004). Google has since claimed to have patched the vulnerabilities announced in this paper, but has not discussed what steps were taken to ensure this. Google has also maintained that there is no evidence to suggest that these vulnerabilities were exploited (n.a., 2005).

A Deeper Understanding of Google Desktop Search

The first point of interest is that Google Desktop Search is only designed for use on NT-based Operating Systems from Windows 2000 and onwards. This may be seen to be isolating a significant portion of potential user-base, but, as discussed below, the program itself makes use of libraries only available in these, newer platforms.

Google has also designed their desktop searching utility to allow third-party additions to its software, publishing several APIs that it uses, allowing for customization in searching parameters. However, all third-party additions must use the Google API to customise settings through the Google program, meaning that direct communication with the database used to store files is not permissible.

Google Desktop Search is comprised of three executables, GoogleDesktopIndex.exe, GoogleDesktopSearch.exe and GoogleDesktopCrawl.exe. GoogleDesktopSearch.exe is the main program of the Google Desktop Search suite, controls the user interaction, and launches the other executables. The GoogleDesktopSearch.exe is the main executable, and operates by setting up a HTTP server on local port 4664. It is from this that all user interaction occurs. GoogleDesktopCrawl.exe is a program that traverses the file structure of a hard disk and reports changes to the GoogleDesktopIndex program. GoogleDesktopIndex.exe interfaces with the persistent storage files, GoogleDesktopCrawl and the Microsoft's Indexing Service. The Indexing Service can send notifications when files are changed, and by listening to this, GoogleDesktopCrawl is able to determine files that potentially require updating.

The Google Desktop Search program creates a registry key at HKEY_USERS\<>SID>\Software\Google\Google Desktop where <SID> is the unique SID, which may look similar to S-1-5-21-3721486523-3945230961-2495595618-1004. There are several options here, including the location for storage of files.

Opening Google's Files

Upon installation, Google Desktop Search makes two folders. The first of these, with the default location \Program Files\Google\Google Desktop Search stores the executables and DLL files required to run the application. The other, with the default installation \Documents and Settings\<>username>\Local Settings\Application Data\Google\Google Desktop Search (where <username> equates to the user that installed the application) stores a series of files named dbc2e.ht1, dbdam, dbdao, dbeam, dbeo, dbm, dbu2d.ht1, dbvm.cf1, dbvmh.ht1, fii, fii.cf1, fiih.ht1, hes.evt, outlook_data, rpm.cf1, rpmh.ht1, and sites.txt. These files are not always present, and there are also several temporary files that are used by the program. Of these, several are human readable, but the majority are not. The file sites.txt is merely a list of different Google mirrors (for example, google.com, google.com.au, etc). The files dbdam, dbdao, dbeam, and dbeao are text-based, and appear to show the process of GoogleDesktopCrawl, and represent all files indexed and websites visited. The non text-based files within this folder are of interest as they may contain the information collected by the Google Desktop Search, the settings for the database used by the program and possibly other data required for the program, such as information required for the Outlook email program (such as passwords to offline folders). It has been surmised that these files are encrypted and/or compressed (Krishnan, 2004). Evidence of compression to these files can be obtained from analysis of the libraries that each of the Google executables utilise.

Using a non-invasive file activity monitor, such as Filemon (www.sysinternals.com), the files and libraries used by processes may be examined. Upon activation, GoogleDesktopIndex.exe calls a series of DLL libraries within Windows. Of these, notably RSAENH.DLL, CRYPT32.DLL, CRYPTUI.DLL and MSASN1.DLL are used for the encryption and decryption of files. Also, the Google installation folder contains gzlib.dll, which is a compression library (Krishnan, 2004). As the Google Desktop Search application does not obviously use encryption for other purposes, and the examination of compressed ZIP files is done via Microsoft's own zipfldr.dll (with the path C:\Windows\System32\zipfldr.dll), the most obvious explanation would be that stored files are encrypted and possibly compressed, as this also accounts for the lack of obvious structure found. Further evidence of encryption is given by the Forensic Tool Kit program (www.accessdata.com), which utilises an 'Entropy Test', designed to detect files which are encrypted, compressed or otherwise obfuscated. Of these files, only the outlook_data file is classified as an encrypted or compressed files – however, testing for entropy will

only indicate files which are entirely encrypted. Based on this, it could be inferred that Google Desktop Search may use a database, the files for which are not encrypted, but all data contained within them may be.

As the Google Desktop Search provides its interface through HTML pages in the default browser, it was hoped that the use of a passive network sniffer, such as Ethereal (www.ethereal.com) could be used to determine the exact communication between the two programs. However, these programs do not monitor the localhost interface, and can only be used in conjunction with actual network connections.

There are several obstacles that need to be overcome before data can be extracted from the Google Desktop Search.

As discussed, the majority of files used by Google Desktop Search are not stored in a human-readable format.

Google Desktop Search makes use of some encryption and possibly compression libraries, and it is not known how these are implemented or how to retrieve this information.

Considering that these files are not readily available to interpretation, one method to view their contents is to use the Google Desktop Search program itself. However, there are several reasons why this is not an optimal solution within a forensic investigation. The first reason is that access to raw data is preferable to information that has been filtered in an unknown way, which the Google Desktop Search program may do. Access to the raw data is much preferred, as it eliminates any contamination which may result from the use of an interface. Searching through the Google Desktop Search interface is also disadvantageous for its inefficiency – the data cannot be ‘browsed’ for information, only searched on specific criteria. This implies that an investigator must already have search combinations in mind before searching for data.

There are also logistical problems with the use of using one copy of the Google Desktop Search program to view files created in another in a forensically sound manner, as this program was never designed to do this. The first obstacle is that although Google Desktop Search has separate programs executing different tasks of the suite, these are inter-dependent and rely upon each other to work correctly. For example, when loading `GoogleDesktopSearch.exe`, the program immediately executes `GoogleDesktopCrawl.exe` and `GoogleDesktopIndex.exe`. If the `GoogleDesktopIndex` process is ended by the Windows Task Manager, `GoogleDesktopSearch` will automatically re-execute it.

What is required is a method of searching the Google Desktop Search program without it indexing or changing files. Google has one solution to this – from within the program a user has the ability to ‘Pause indexing’. This action pauses the `GoogleDesktopCrawl` program. However, this occurs after Google Desktop Search is running and presumably indexing, so it occurs too late. Ensuring that the files required by Google Desktop Search are read-only (either by changing the default storage location in the registry to a CD media or by changing the attributes) is also not effective, as the program performs a check on this before executing. When loading the files, if they are not able to be written to, the program fails with Database error 13387, which automatically diverts to Google’s help centre.

One method to prevent Google Desktop Search from indexing at all is to prevent the two components of the program responsible for indexing and updating the cache from loading, by manually renaming both the `GoogleDesktopCrawl.exe` and the `GoogleDesktopIndex.exe` executables. This prevents them from being activated by the original program when the program is first loaded. However, as these tools are so inter-dependent, running `GoogleDesktopSearch` independently of the other two programs results in only the Google Desktop Search icon in the taskbar – no other functions of the program operate correctly. It would appear from this that the `GoogleDesktopIndex` operates components of the user interface. However, by renaming only the `GoogleDesktopCrawl.exe` (for example, renaming it to `GoogleDesktopCrawl.exe2`) solves many of these issues. The program will still execute and the user interface is still accessible, but the indexing of files does not occur.

One must also be careful about Google Desktop Search creating and altering files whilst in operation. Whilst the authors have been unable to reproduce the exact conditions under which this occurs, the files that are created are temporary and removing them does not affect the integrity of the results produced. Similarly, the `outlook_data` file produced will be altered by an open copy of the Microsoft Outlook program. Google Desktop Search also will edit all files contained within the default storage location when it is manually closed, with the exception of the `dbdao` file.

From this, there can be derived a procedure for viewing the stored contents of the Google Desktop Search program without tampering with them:

Copy the Google Desktop Search storage folder (where the default is `c:\Documents and Settings\<username>\Local Settings\Application Data\Google\Google Desktop Search`) from the source machine to the Google Desktop Search folder on a machine conducting analysis

On the analysis machine, rename the file `GoogleDesktopCrawl.exe` to `GoogleDesktopCrawl.exe2`. This will prevent it from loading.

Open the Google Desktop Search program, ensuring that no Email programs are loaded.

After the Google Desktop Search program has loaded, traverse to the storage folder on the analysis machine, and change the file attributes of these files to Read-Only. This will allow the Google Desktop Search program to close without editing any files.

THE USES OF GOOGLE DESKTOP SEARCH WITHIN FORENSIC COMPUTING

Although the storage files of Google Desktop Search are not human-readable, the data that is stored within these files is still accessible, although access to the data is limited to the Google Desktop Search user interface.

Searching and storage of emails is a varied task, as it depends on the type of mail used and how the client has been configured. In the case that email is stored remotely via an IMAP (www.imap.org) or through the Exchange protocol, it may be problematic or time-consuming retrieving all email from a machine. However, Google Desktop Search stores emails locally for searching, which is accessible through the program. This includes offline storage such as Microsoft Outlook's use of .PST files to store information.

By far the most unique feature within Google Desktop Search for a Forensic investigator is that the program caches, indexes and stores Internet sites visited, much in the same way that Windows does, by default. This is the only desktop searching utility with this feature, and possibly stems from Google's background within the Internet searching field. Google Desktop Search performs all cataloguing and indexing entirely independently of the Windows caching of Internet pages, so should a user delete their temporary Internet files, cache and cookies, this record is maintained by the Google Desktop Search program. Google Desktop Search caches all HTML Internet pages visited, including pages retrieved via an SSL connection (this can be removed via a configuration option). This has added benefit when it is realised that there are several programs available designed to remove this very information in an irretrievable manner, but these operate solely with the Operating System, and fail to take into account any other programs that may be collecting and storing this data. Additionally, should a single webpage have been visited repeatedly, the Google Desktop Search will store cached copies of all of these pages, giving exact information on what was presented to the browser on each occasion visited. Much in the same way that the Google Internet Search (www.google.com) caches popular pages, only the HTML is stored with images retrieved from the remote site.

Whilst the program does not store images locally, either from local or remote locations, it often will store thumbnails of images that are stored locally on a system. This is independent of the image itself – not arranged on the fly, meaning that investigators interested in images that may have been altered or deleted may still find a thumbnail PNG file 109x75 pixels in size.

The Google Personal Desktop Search was remarkably interesting for its caching of certain file types such as text, that continue to exist after the original item has been deleted. This may continue indefinitely, and the result is not easily removed.

LIMITATIONS OF DESKTOP SEARCH UTILITIES

As Desktop Searching programs are primarily designed for users to locate files, images, emails or Internet history, forensic analysis of metadata produced by these programs may not provide an accurate representation of the files contained with these machines. This is intentional in the programs designs, as they are designed to index and retrieve user-created data, and will therefore not index all files on a machine, merely ones that conform to particular criteria and are stored in locations that are likely to contain such data. Google Desktop Search did not search or index all files, but narrowed search space to areas that are more liable to contain documents stored by the user rather than files used to operate and maintain the machine. Files stored within the default Windows directory, within the Recycle Bin or that are invisible were not be indexed, as it was unlikely that these areas would yield results, and their exclusion increases the efficiency of the program.

These restricted searching limits the results returned and stored by Desktop Searching programs and reduces the impact that analysis can provide, as it is possible to ensure that should these files exist on a given machine, they are not indexed.

It would be a simple matter for a user to ensure that particular files stored on a machine are not searched and indexed by a Desktop Search program, but these programs are not designed for thorough searching, rather to aid the user where appropriate. From a forensic computing perspective, it cannot be assumed that any data found within these programs could be considered complete, as it is a simple matter to ensure that files are not indexed. The benefit here does not lie in providing a complete account of all activity in itself – merely another source of potentially enlightening material.

The increased usage of utilities that provide metadata for a particular system beyond that created by the Operating System may have several benefits for those in Forensic Computing Investigation, as they may create

data that does not exist in any other form or has been deleted, and may be used to verify other data by providing consistent results. For example, Google Desktop Search retains past Internet history independently of the Operating System and browser, and needs to be cleared independently by the user. Even current 'disk-wipe' programs, designed to securely delete Internet history, recently opened documents and slack space make no claim to removing the metadata produced by these programs.

There are a number of disadvantages to the increased use of Desktop Searching programs, and in their current stage they only have limited applicability. As discussed, one major limiting factor for utilities such as Google Desktop Search is that they have a refined searching field and only index files according to strict criteria of visibility, location and file extension. Further, as these are still new technologies, their interface and searching mechanisms are often primitive and unsuited to the personal desktop. Searches are made by keyword and cannot be made by date or other factor. It is this that limits these programs usefulness, as without a clear indication of what to search for, there is a possibility that information will be missed. Within the Google Desktop Search, a search for a word will not return results with that word as a substring, so a search for 'celeb' will not return results where the word 'celebrity' appears. Whilst this is logical within an Internet search, which may return results numbered in the millions, this closed approach is not suited to a desktop, and when trying to extract information from the stored search data, it is tedious.

The incomplete nature of the Google Desktop Search can be further identified when the process discussed above, to read index data created by other machines and other copies of Google Desktop Search, stops the indexing process. Shutting off the indexing component of the software prevents the program from indexing files changed during this period – it fails to register changes made even once it has been resumed. This could be because the program has failed, rather than being manually shut off from within the program.

As there is to date no single product dominating this market there are several proprietary data formats used for the storage of the data produced by these programs, and tailoring of a system, when possible, is designed to act as a plug-in for an already running program, which is not applicable within a Forensic Computing context. Google Desktop Search makes some use of encryption, compression or obfuscation to ensure that the information collected is not be human-readable, or could not be used by other programs. Whilst this makes good sense from a computer security perspective, it also reduces the effectiveness of these programs from a Forensic Computing perspective, as the raw data is not available for viewing, only through the program interface.

CONCLUSION:

Whilst still new, the desktop search utility represents a growing area of software, with many Internet-based companies adapting their work to this area and merging their services. Where this makes sense for investigators is that these programs often store data independently of an operating system platform, and hence may contain extraneous metadata, which has potential use within an investigation. As these programs become more popular and as they improve, their use will only grow and they will become more powerful.

Discussed here is only a work-around solution to extracting the data stored within Google Desktop Search, whereas ideally, extracting, interpreting and querying the data directly would be a preferable solution. The most obvious method of doing so would be to reverse engineer the storage files, and construct programs to analyse and present directly from the raw data. However, given the use of encryption, this could possibly be a time-consuming task.

A more feasible solution would be to expand on the solution discussed above, and reiterating that Google does allow third part extensions of their work, write plug-ins or programs that utilise the GDS Developer Search API, and performing more exhaustive and in-depth searches. This would not be difficult and would make data-mining much more automatic in nature.

These programs exist only to overcome the limitations found within existing search programs and it is unknown if in the long-term, these programs will continue to exist. Microsoft have released their own searching program, which could potentially be integrated into the next Windows release – codenamed Longhorn. It is not difficult to see this occurring, and if it does, then there will be no need for any other, similar programs. But for the moment, there is a market for these products, and it does provide another source of data that may be of use, as often the user-data captured is similar to the data searched for within a Forensic Investigation.

REFERENCES:

- Krishnan, S, 2004, Reverse Engineering Google Desktop Search, available at <http://dotnetjunkies.com/WebLog/sriram/archive/2004/11/22/33091.aspx>
- n.a., 2005, Google Desktop Search Release Notes, Google, available online at <http://desktop.google.com/releasenotes.html>

Nielson, S., Fogarty, S., & Wallach, D., 2004, Attacks on local searching tools, Technical Report TR04-445, Department of Computer Science, Rice University. Available at <http://seclab.cs.rice.edu>

Posey, B., 2005, The Security Risks of Desktop Searches, Windows Security. Com, available online at www.windowssecurity.com

Spring, T., 2004, Google Desktop Search: Security Threat?, PC World Magazine, October 15, 2004, available online at <http://blogs.pcworld.com/staffblog/archives/000264.html>

COPYRIGHT

Turnbull, Blundell, Slay ©2005. The author/s assign the School of Computer and Information Science (SCIS) & Edith Cowan University a non-exclusive license to use this document for personal use provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive license to SCIS & ECU to publish this document in full in the Conference Proceedings. Such documents may be published on the World Wide Web, CD-ROM, in printed form, and on mirror sites on the World Wide Web. Any other usage is prohibited without the express permission of the authors.