Global Information Assurance Certification Paper

Detecting and Protecting Against Word Field Code Abuse
GIAC Security Essentials Certification, version 1.4b
Option 1 – Research on Topics in Information Security
Mark E. Soderlund
March 6, 2003

Abstract

Many personal computer users today are using versions of Microsoft's Word software to do their word processing. They collaborate on various projects through the sharing of Word documents. In a collaborative effort, trust must exist between all parties. However, a feature of Word used to manage dynamic document data has been demonstrated to allow a person to abuse this trust and "steal" data files and other information from another party. This is accomplished by inserting a series of field codes into a Word document. When the other user reviews, saves and returns the document, the abuser receives the desired information.

This paper describes a new technique, originated by the author, which uses various "off the shelf" tools as a method for finding field code abuse in Word documents on a single computer or an organization's file server. This approach can be used to find evidence of previous field code abuse and aid in its analysis. To this end, field code abuse will be described and various tools will be evaluated for potential use in detecting and protecting against this vulnerability.

Background

On August 26, 2002, Alex Gantman reported on Bugtraq (1) that he had found a way to use a Word 97 document to obtain a file from another person. He called it "Document Collaboration Spyware". User A (the attacker) includes a series of field codes (spy code) in a document and when User B (the target) opens the document, Word automatically includes the file requested in the field codes, if it exists. If User B saves the updated document and sends it back to User A, User A can open it in WordPad or some editor other than Word. He then has access to the included file.

Woody Leonhard, publisher of "Woody's Office Watch" and other Windows related e-mail newsletters, warned his readers of this vulnerability in his September 6, 2002 newsletter (2). Over the next few issues of his Office Watch and Windows Watch newsletters, Mr. Leonhard introduced a tool to "sniff out" misuses of fields (3) and a "Hidden File Detector" Word add-in (4). He also stated that multiple spy codes can be included in a single document, or that spy codes can be crafted to send files or other information to a web site (5). Spy codes can also be crafted that use dynamic data exchange (DDE) links to obtain information from other Windows programs.

Microsoft responded to the original notice of the vulnerability on September 13th 2002(6), and on October 16th they posted a security bulletin MS02-059 (7). That bulletin pointed readers to a patch for Word versions 2002, 2000, X for Macintosh, 2001 for Macintosh, and Excel 2002. The security bulletin referred Word 97 & 98(J) users to Microsoft Product Support Services since those releases are only supported through assisted support. In other words, the patch for Word 97 and 98(J) is not available for free download. After Microsoft released their patch, Mr. Leonhard introduced a "Field Security Thermostat" utility to help manage the patch's actions (8).

Microsoft's patch and the other tools mentioned above help deal with individual cases of abuse. However, they do not provide assistance to a security manager who needs to determine whether any of the thousands of Word documents that are stored on his organization's hard disks contain spy codes. The only method available is to check every Word document individually. A method of narrowing the focus of a search to potential suspect documents is needed.

How Field Codes Work

Understanding this vulnerability requires a basic knowledge of field codes, their purpose and how they are updated. Microsoft Word's help says "Fields are used as placeholders for data that might change in a document and for creating form letters and labels in mail-merge documents." Anyone who has used the Insert menu and selected "Date and Time" or "Page Number" has used Word fields. They also control other important Word features like the creation and maintenance of a table of contents. Fields are generally added to a document by selecting the "Insert > Field" menu item or by typing Ctrl+F9. Most fields are updated only when the user selects the field and requests it be updated, either by right clicking and selecting the Update Field menu item or by typing the F9 key. By default, fields are hidden from view, but they can be revealed either by typing ALT+F9 or by selecting the "Tools > Options" menu item, selecting the View tab and selecting the Field Codes check box.

In an attempt to learn more about fields, a search of the Internet turned up the document "Microsoft's Word Field Codes: Revealing Their {Private} Side" (9) by Sherry Kappel. It describes the purpose of field codes, how they are entered and updated in documents and classifies them by their purposes. It also contains Visual Basic code to search for and manipulate field codes.

Fields are separated from normal text in a Word document by using the opening and closing field characters, shown onscreen as "{" and "}" respectively. The opening field character is hexadecimal 0x13 (19 decimal), and the closing field character is 0x15 (21).

The question "How are fields updated?" is best answered in Ms. Kappel's writing (9). She states:

> "There are three kinds of field codes in Word: **hot field codes, warm field codes and cold field codes**. These kinds of field codes differ only with

> respect to their update requirements: hot field codes need not be updated by the user; warm field codes must be updated by the user using the **F9** key; and cold field codes cannot be updated using the **F9** key, at all: they must either be reinserted, or their 'contents' must be edited by the user."

Based on this information, the only fields that should update automatically are the hot kinds. Ms. Kappel also lists all fields available in Word and their "kinds". Mr. Gantman (1) found that the last DATE field in a Word 97 or Word 2000 document is always updated when the document is opened. The DATE field code is normally a "warm" kind of code, except in this one condition.

Fields are also updated when a document is printed. The "Tools > Options, Print tab, Update fields check box" controls whether all fields in a documents are updated when printed.

Fields can be nested, and evaluation in those cases works from the innermost field to the outermost. Mr. Woody Leonhard presents a detailed walk-through of the construction of a spy code and its evaluation in his "Woody's Office for Mere Mortals" newsletter of 26 September 2002 (10).

Risk Assessment

As with all potential threats, an organization should analyze the risk this vulnerability presents to their environment and act accordingly. SANS defines risk as being equal to vulnerability times threat. One organization may consider this to be a serious threat, while another considers it trivial. Microsoft classifies this vulnerability as a moderate risk (7). The US Department of Energy Computer Incident Advisory Capability (CIAC) considered it a low risk in their bulletin on the vulnerability (11). A security officer needs to ask the following questions to determine their organization's risk.

> Are we using Microsoft Word and Excel?
> Do we use Word and Excel to collaborate internally or externally?
> Can we apply the available patches, or will the patches cause internal processes (mail merge) to fail?
> Is the risk of having an internal user or external user obtaining information they are not authorized to have acceptable?

Organizations that use WordPerfect Office or Lotus SmartSuite exclusively have no vulnerability - therefore, no risk. The same is true for one that does not collaborate by sharing documents. After that, the level of risk must be individually determined.

Microsoft stated four points in determining the vulnerability's severity in their statement (7).

"The attacker would need to know the absolute path to the file that is to be stolen.
The attacker would need to entice the user into returning the document.
The user could always view the field codes.
The attacker would leave a clear audit trail."

The first point was shown to be false as the fields are able to search the directories in Word's "Tools > Options, File Locations" setting. Therefore, a file in the default "My Documents" folder is vulnerable without the attacker knowing the exact path to it. Also, many companies have a policy of using fields to put the filename of documents inside their documents to assist in later retrieval. An attacker who has received a printed document from one of these companies would have a good start on where to look. Microsoft's statement ignores the fact that the greatest risk to information security in an organization comes from people inside the organization, not outside of it. In this case, the user's knowledge of how information is stored could negate this argument. Also, the common practice of using fields to include a file's storage location in the file itself would provide recipients of these documents with information that could be used to exploit this vulnerability.

Many applications have "default locations" where they store information. For example, the Quicken financial program uses the file C:\Quickenw\Qdata.qdf to store its information. Windows NT, 2000 and XP systems that are either stand-alone systems or Workgroup members store the data for their Security Account Manager in the file "*%systemroot%*\System32\Config\SAM". Crafting a spy document to obtain this file would allow an attacker to use password cracking tools to gain access to these systems. Even if the attacker was not sure whether the data file was stored on disk C: or disk D:, the spy document could be crafted to check both disks for the file, or even all drives from C: through Z:.

Concerning the second point, getting a user to return the document is not a major issue in today's environment of network storage and collaboration. Reviewing and commenting on a document is part of normal business for many computer users today. A well crafted request to the target could be just as effective in getting a response as the social engineering used by the author of the "I Love You" virus. The inclusion of obvious errors into the spy document would increase the likelihood of the target returning the "corrected" document. Tests also show that the attack document can be crafted to send about 230 characters to a web site the attacker has set up. This does not require the document be returned.

The third point is true. All field codes in a document can be revealed simply by typing ALT+F9, or by selecting the "Field Codes" view option described earlier. If the spy code is in a footer or header, those need to be made visible by selecting "View > Header and Footer". If an attacker used spy codes to send information over the web, the information has already been delivered before a user reviews

the fields, but review of the fields would let the victim know this has occurred so he could act on this information in a timely manner.

Sherry Kappel stated this very point in 2000 when she wrote "Because we are trading Word documents with the rest of the world these days, it is important to realize the scope of Word's field code feature set, understand how to locate and identify them, how to evaluate their use, health and appropriateness in the document, and how to manipulate them, ..." (9).

This fourth point is true. As long as the user has a copy of the document, there is evidence that can be used to find out what information has been taken. The potential of being caught may be sufficient to keep many people from taking advantage of this vulnerability. A spy document on a corporate network could be modified as soon as it has been used to steal data, thus removing the evidence. Knowing the frequency of backups would allow the attacker to delete or replace the spy document with a "clean" version. Having evidence of the attack would be useful in any legal action that could ensue. This document described one method which could be used to aid in a forensic analysis of an incident.

Protecting from Field Code Abuse

Protection from this vulnerability requires some understanding of field code abuse. The original Bugtraq posting (1) included the following example.

> "Proof of concept:
> Inserting the following field structure into the footer of the last page will steal the contents of c:\a.txt on the target's computer. Keep in mind the plain curly braces below must actually be replaced with Word field braces (you can either use the menus to insert fields one by one, or ask google how to do it by hand).
> { IF { INCLUDETEXT { IF { DATE } = { DATE } "c:\\a.txt" "c:\\a.txt" } * MERGEFORMAT } = "" "" * MERGEFORMAT }"

The field being used in the above case is INCLUDETEXT. Microsoft's patch manages the updating of the fields DDE, DDEAUTO, INCLUDEPICTURE, INCLUDETEXT, IMPORT and DATABASE.

The Hidden Files Detector allows a user to check a document for spy codes while reviewing a Word document. After installing the add-in, it is available by selecting "Tools > Detect Hidden Files" (Figure 1). All potential spy codes are displayed in the window and the "Go to" button will move to the suspect field and reveal the fields (Figure 2). When Hidden File Detector is installed, a Word document "Sample Spyware.doc" is also installed to provide users with an example of a spy code.

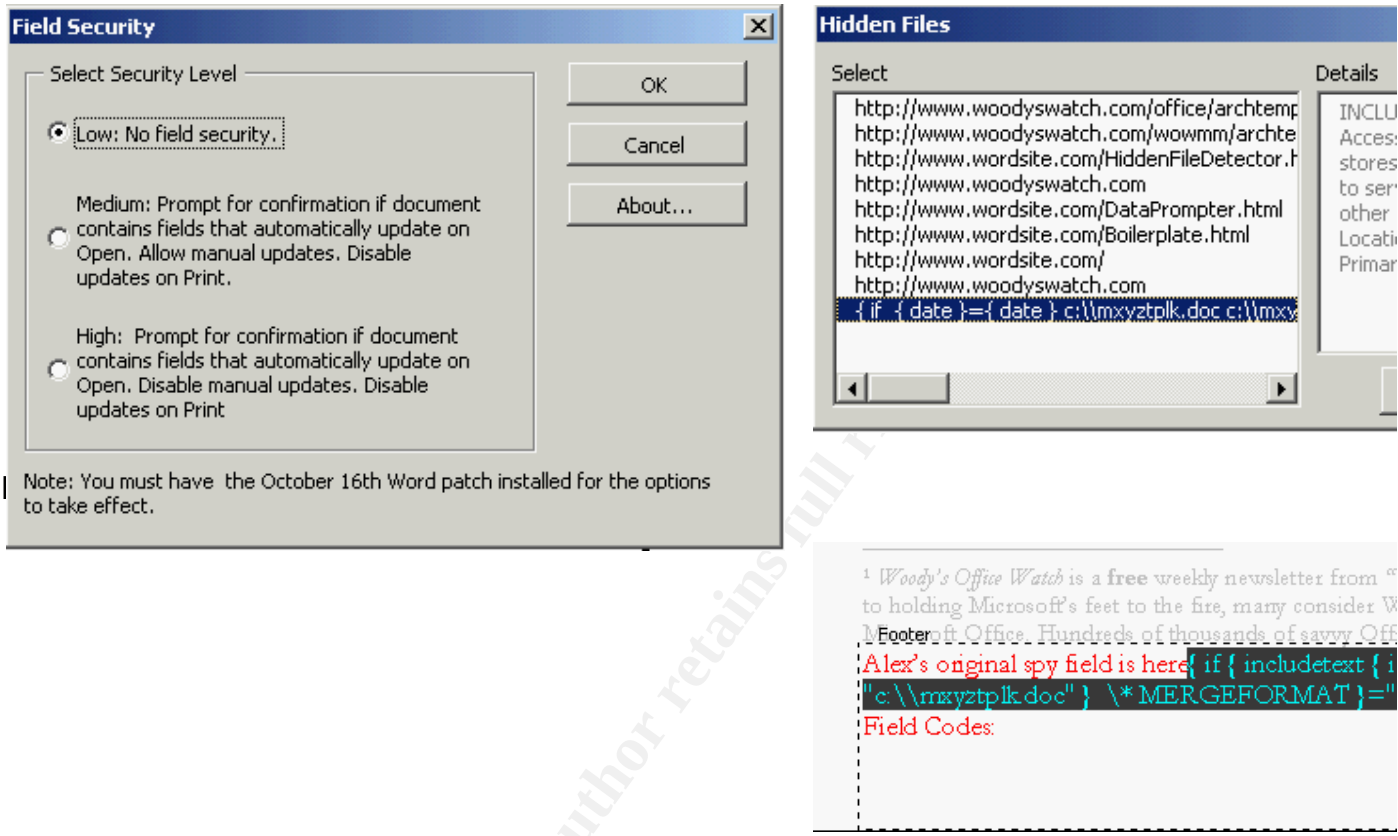


Figure 2 - Field Codes Revealed in Sample Spyware.doc

The next step was to install and test Microsoft's patch and the Field Security Thermostat. The above tests have been performed with Word 2000, so the patch for that version was tested. The patch comes as an executable file and installed easily, although it did require the Office 2000 CD-ROM during the installation. The patch makes Word check the value of the registry key "FieldCalcSecurityLevel" in HKEY_Current_User\Software\Microsoft\Office\9.0\Word\Options. The key has three valid values, "0", "1" and "2". A value of "0" maintains the original vulnerable behavior of automatically updating fields that can be used for spying. A value of "1" means Word will prompt the user before updating these fields. A value of "2" means these fields cannot be updated. The registry entry is read during Word startup, so changes to it require stopping and restarting Word. Since many computer users are not comfortable editing the registry, Beth A. Melton of Melton Consulting developed a Word add-in to facilitate changing the setting (Figure 3). Woody Leonhard refers to her tool as the Field Security Thermostat and has included it with the Hidden File Detector. He advertised its availability in his newsletter in December 2002 (6). After installing it, the user can choose the "Tools>Change Word Field Security" menu item and select the appropriate security level. After installing the Field Security Thermostat, the Detect Hidden Files option is no longer visible on the Tools Menu. Selecting the "Tools > Templates and Add-Ins" menu item and unselecting the Field Security add-in

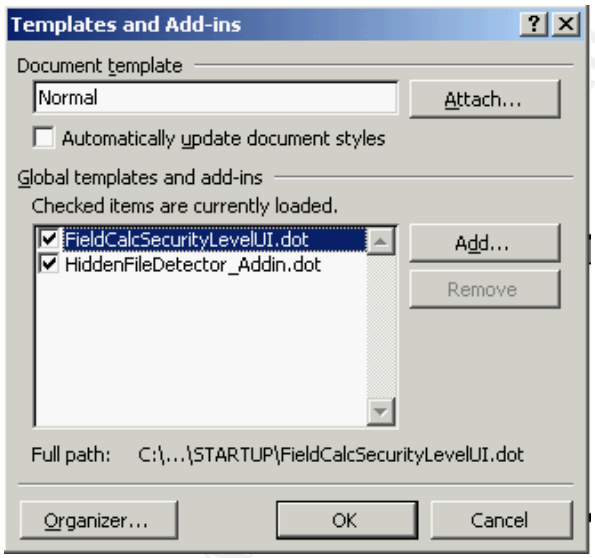
(Figure 4) allows access to the other add-in. No method was found to allow both add-ins to appear on the Tools menu at the same time. This problem exists in Word 97 and 2000.

These Word add-ins are installed in different places depending on the version of Word being used. With Word 97, they are installed in the directory "*Drive:*\Program Files\Microsoft Office\Office\STARTUP". This means they are installed for everyone who uses that PC. With Word 2000 and later versions, the add-ins are stored in the individual user's profile, i.e. "C:\Documents and Settings*Word User*\Local Settings\Application Data\Microsoft\Word\Startup".

Figure 4 - Word Add-in Control Screen

Copying the add-ins into the "All Users" profile did not make them available for everyone who uses the computer. This means that these add-ins need to be installed for each individual who will use them on that PC. If the organization does not use "roaming profiles", each user will have to install the add-ins on every PC that they use as well.

Figure 3 - Field Security Thermostat



Searching Multiple Files for Field Abuse

With the installation of Microsoft's patch and use of the Hidden Files Detector add-in, a Word user is now able to block unscrupulous activities from occurring and check for dangerous field use in a document. But what should be done about the numerous Word documents that are already stored on hard disks and file servers? How should they be checked for these codes? The task of scanning every Word document using Hidden File Detector is simple when dealing with a small number of documents. However, organizations may have hundreds to thousands of documents to check. What is needed is a way to search all documents on an individual computer or a network for possible spy codes, and then to use Hidden File Detector to check the files found by that search. The computer on which this document was written has over 300 Word documents on its hard disk. If one document could be scanned each minute, over 5 hours of time would be required to search this one PC. A typical file server that a security officer needs to check could easily hold tens of thousands of documents.

A search tool which is able to find one of these fields can be used to find the others. Microsoft includes the ability to search files for text in its Windows operating systems. The search capability in Windows is limited, as it will stop when it finds 10,000 matches to its search criteria. Using the "Sample Spyware.doc" file that comes with the Hidden File Detector as a target, search tools were evaluated to determine their ability to locate the INCLUDETEXT field. Windows search was started (F3 key or "Start > Search > For Files and Folders") and the string "INCLUDETEXT" (ignoring case) was searched for in the directory containing "Spyware Sample.doc". Even though this code was present, the search function could not locate it. Windows find was ruled out as a possible search tool.

Determining how the field code "INCLUDETEXT" was stored in the document required looking at the file as raw data. A hex editor is designed for this very purpose. The freeware tool "Hex Editor v 2.0" from HHD Software (www.hhdsoftware.com) is one of many hex editors available for download and was used in the preparation of this document. Searching the spyware sample document revealed the INCLUDETEXT field – with NULL characters (ASCII 00) between each letter. (Figure 5).

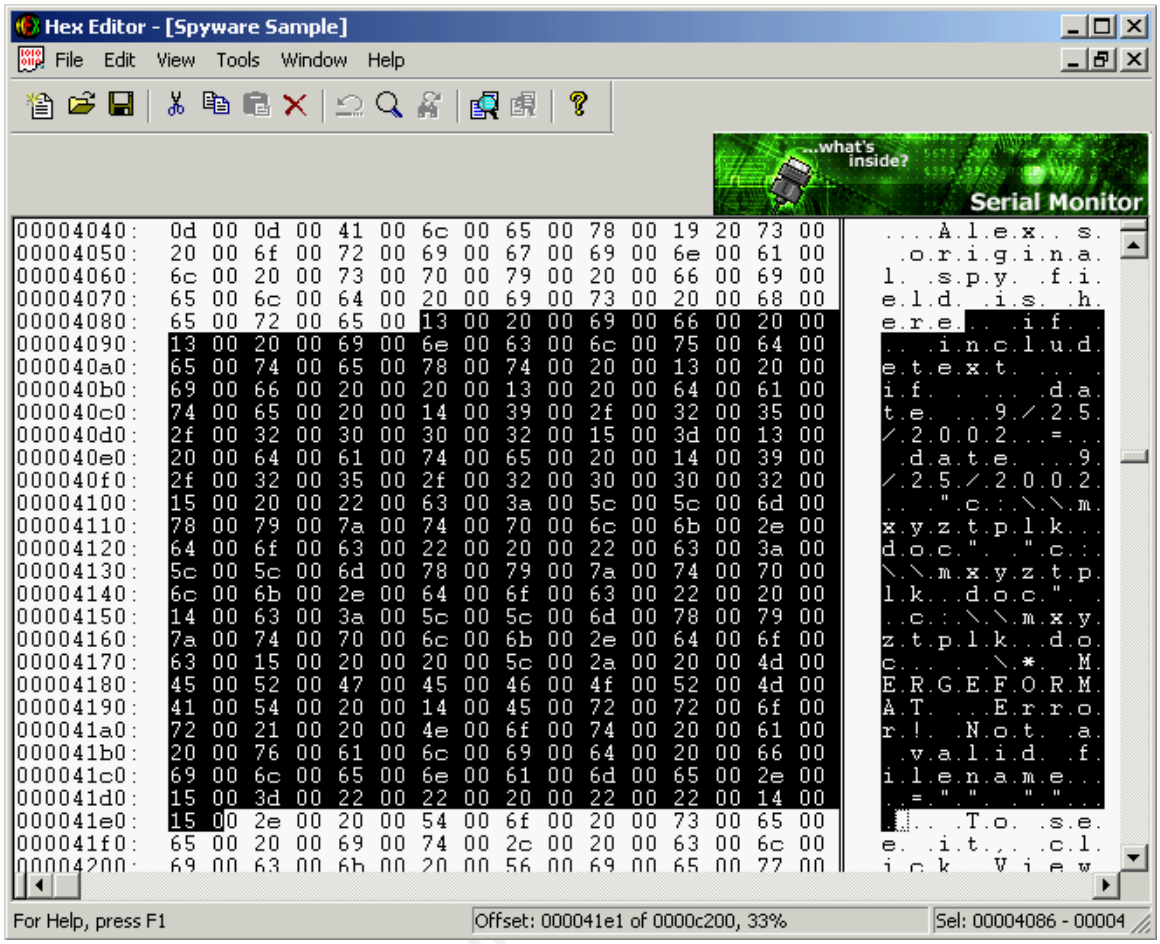


Figure 5 - Hex Editor Screen with Spy Code Hi-lighted

Attempts to use the find function of Windows to search for the string with the embedded NULLs were unsuccessful. Some other search tool would have to be found that could search for binary data in text. The UNIX "grep" command (global regular expression print) is able to find binary data, so a Windows version of grep might be able to replace Windows find. A version of the VI editor, WinVi32, was used successfully to test the assumption that regular expressions could locate this field code. This method still required that each file be checked individually. A tool that could search for text with embedded binary characters that could also recursively search sub-directories was needed. A visit to Download.com, produced a list of several freeware and shareware search utilities that might be able to meet these requirements. Although the site listed several tools that advertised "grep functionality for Windows", most attempts to use the regular expression that worked in WinVi32 failed.

One program tested which was able to locate the INCLUDETEXT field code, was Search and Replace, a shareware program from Funduc Software (www.funduc.com). Search and Replace has a special binary mode for searches of this kind. The search string can be typed in or the characters can be selected

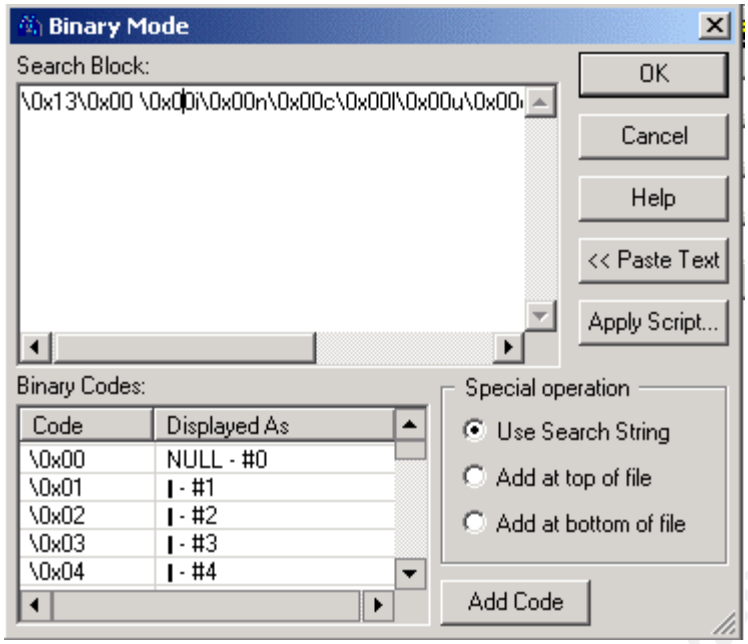


Figure 6 - Search and Replace Binary Mode

from the list in the bottom of the window (Figure 6). The files found (Figure 7) can be opened with Word and the Hidden File Detector run to determine whether their use was benign or malicious. Search and Replace was used to search for a spy code in another document to verify that it would be able to find spy codes consistently. A new document was created with the INCLUDETEXT field code and the search was repeated. This time Search and Replace was unable to locate code in the new file. Checking the new file in the Hex Editor revealed that Word had not included the NULL characters this time. The Windows find function was able to find the spy code in the second example, but not the first. The existence of two options presents the possibility of other options. Therefore, at least two searches would be needed to find documents that required further checking.

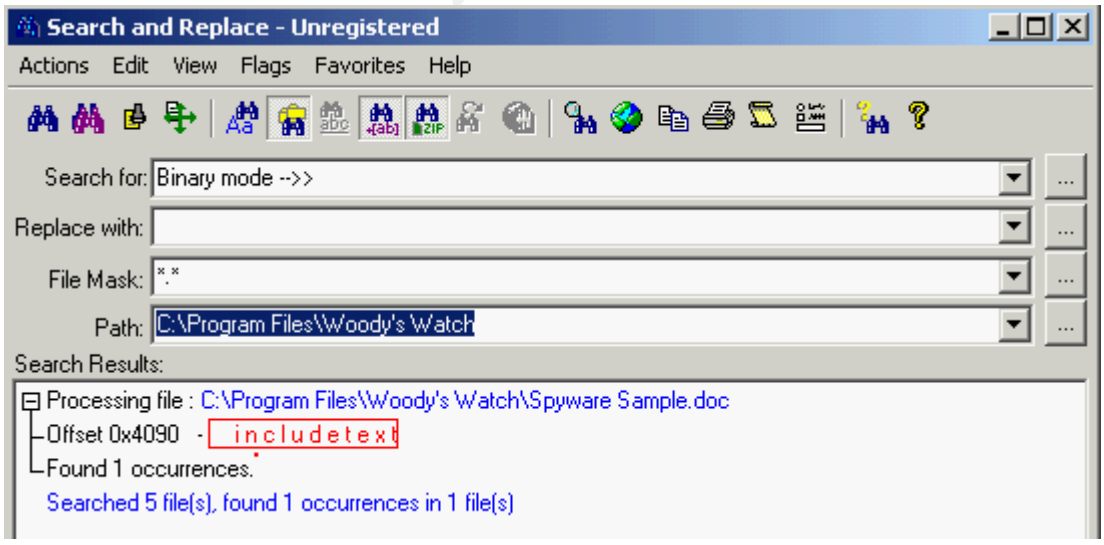


Figure 7 - Search and Replace - Search Results

A search tool was needed that understood Word's document structure. Testing of some of the other downloaded search programs revealed that Agent Ransack, a

freeware program by David Vest (www.agentransack.com), has that capability. When given the plain text search string, it found both example files (Figure 8).

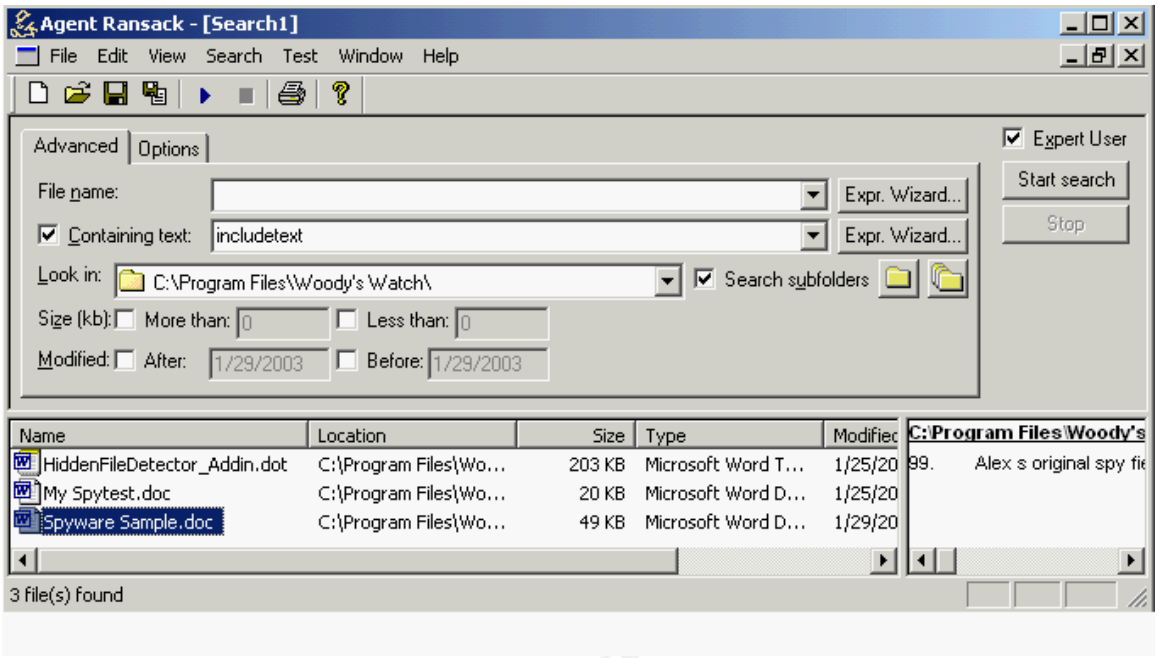


Figure 8 - Agent Ransack - Search Screen

With Agent Ransack, a tool to search hard disks and networks for possible spy codes had been found. It is able to use "logical ors" in searches (using the "|" character), so a search for all the possibly misused codes could be constructed, eliminating the need to change the search string and repeat the search. Recently used search strings are available in a drop down list which could eliminate the need to type them in repeatedly. Files found during the search can be selected and opened in Word to be checked by the Hidden File Detector. This search program is freeware, a bonus to small companies and those with tight budgets.

The set of utilities provided by Agent Ransack and the Hidden File Detector was tested on a real world network storage location. Agent Ransack searched over 10,000 files and found only two that used the INCLUDETEXT field. Both were checked using Hidden File Detector. The field in both cases was used to include an appendix to a document. No field abuses were discovered in this example, but the ability of these tools to find the field proved the concept.

When using any search tool, care must be exercised in defining the search strings. A test using all of the fields that the Microsoft patch manages revealed 27,830 files with matches in the same network storage location used above. The string "DDE" was found in several files that did not contain that field code, but the word "bidder". The fields "IMPORT" and "DATABASE" can easily appear in a normal document. Placing spaces around the strings might eliminate these "false positives" because Word automatically puts a space before and after the field

when one is created. Also, using the ability of Agent Ransack to use regular expressions, it can look for these common strings with a non-alphabetic character (not a through z) before them. This regular expression is "[^a-zA-Z] database". Another approach might be to create a program to search for these fields and only report them when they are preceded by the special character ({) Word uses to indicate the start of a field. Creation of such a program was outside of the scope of this paper.

Since collaboration often occurs through exchange of e-mail attachments, evidence of spy codes may exist within an organizations e-mail server's databases. Discovering a method for searching for spy codes within these databases was outside of the scope of this paper.

Conclusion

The abuse of Word's field codes is a simple method an attacker might use to gain information to which they would not normally have access. A clear audit trail and the ability to determine that information has been obtained through this vulnerability is no substitute for vigilance. It is better to prevent the type of abuse described from happening than to know that it has occurred. The principal of defense in depth would suggest several courses of action. First, a policy of not reviewing documents from other sources without scanning those documents for potential abuses should be adopted. Second, the installation of the patch from Microsoft should be a priority for Word users who do not routinely use field codes in documents. After installation, the registry value should be set to 1 to prompt users before fields are updated. Users of "mail merge" and other dynamic documents will want to obtain the other tools discussed above as well, so that they can be in control of what information is included in documents. Finally, security training for users will help them know what tools are available to help prevent this kind of abuse and its consequences. The Hidden File Detector, Field Security Thermostat and Microsoft's patch provide individual users the tools they need to protect themselves from this vulnerability. The addition of Agent Ransack to this set of tools allows security officers and network managers to evaluate their areas of responsibility for evidence of exploitation of this vulnerability.

References

1) Alex Gantman, "Security side-effects of Word fields", 26 August 2002, http://online.securityfocus.com/archive/1/289268.
2) Woody Leonhard, "WOW #7.42 - The Biggest Word 97 Security Hole Yet?", 6 September 2002, http://www.woodyswatch.com/office/archtemplate.asp?v7-n42.
3) Woody Leonhard, "WOW #7.44 – a Field Sniffer now available!", 20 September 2002, http://www.woodyswatch.com/office/archtemplate.asp?v7-n44.
4) Woody Leonhard, "WOW #7.45 – Your Word 'spy' questions answered.", 26 September 2002, http://www.woodyswatch.com/office/archtemplate.asp?v7-n45.

5) Woody Leonhard, "WOW #7.47 – Lies, spy codes, Xdocs and more lies", 9 October 2002, http://www.woodyswatch.com/office/archtemplate.asp?v7-n47.
6). "Information about Reported Microsoft Word Fields Vulnerability", http://www.microsoft.com/technet/security/topics/secword.asp, Microsoft Corporation, 13 September 2002.
7) "Microsoft Security Bulletin MS02-059", http://www.microsoft.com/technet/security/bulletin/ms02-059.asp, Microsoft Corporation, 16 October 2002.
8) Woody Leonhard, "WOW #7.55 – Free Field Security Manager", 11 December 2002, http://www.woodyswatch.com/office/archtemplate.asp?v7-n55.
9) Sherry Kappel, "Microsoft's Word Field Codes: Revealing Their {Private} Side", http://www.microsystems.com/Fields_Revealed.htm, Microsystems Engineering Company, 6 June 2000.
10) Woody Leonhard, "WOW-MM #3.23 Get your hidden file detector here!", 26 September 2002, http://www.woodyswatch.com/wowmm/archtemplate.asp?v3-n23
11) US Department of Energy Computer Incident Advisory Capability, "CIACTech 02-005: Understanding Capturing Files with Microsoft Word Field Codes", 27 September 2002, http://www.ciac.org/ciac/techbull/CIACTech02-005.shtml.