E-mail Security in the Wake of Recent Malicious Code Incidents

By: Trent Pitsenbarger
and
Paul Bartock
of the
Systems and Network Attack Center (SNAC)
W2Kguides@nsa.gov

Acknowledgments:
The authors would like to acknowledge
Neal Ziring and Dave Albanese, NSA and
Sean Finnegan, Microsoft for their contributions.

Dated: Jan 29, 2002
Version 2.6

Warnings

- **Do not attempt to implement any of the settings in this guide without first testing in a non-operational environment.**
- This document is only a guide containing recommended security settings. It is not meant to replace well-structured policy or sound judgment. Furthermore this guide does not address site-specific configuration issues. Care must be taken when implementing this guide to address local operational and policy concerns.
- SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED. IN NO EVENT SHALL THE CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
- Please keep track of the latest security patches and advisories at the Microsoft security bulletin page at http://www.microsoft.com/technet/security/current.asp.
- This document contains possible recommended settings for the system Registry. You can severely impair or disable a Windows NT System with incorrect changes or accidental deletions when using a Registry editor (Regedt32.exe or Regedit.exe) to change the system configuration. Currently, there is no “undo” command for deletions within the Registry. Registry editor prompts you to confirm the deletions if “Confirm on Delete” is selected from the options menu. When you delete a key, the message does not include the name of the key you are deleting. Therefore, check your selection carefully before proceeding.

Trademark Information

(U) Microsoft, MS-DOS, Windows, Windows NT, Windows 98, Windows 95, Windows for Workgroups, and Windows 3.1 are either registered trademarks or trademarks of Microsoft Corporation in the U.S.A. and other countries.

(U) All other names are registered trademarks or trademarks of their respective companies.

Table of Contents

Introduction

The recent spate of malicious code based attacks, most recently exemplified by the ILOVEYOU worm, has highlighted the propensity of modern e-mail systems to provide a ready conduit for malicious code delivery. The Microsoft family of e-mail clients has proven to be a particularly attractive target for malicious code writers, primarily due to their widespread usage and their rich programming model.

While there have been numerous malicious code payloads that have targeted the Microsoft environment, three standout given their impact or the varying approaches they utilized. The Melissa virus delivered its destructive payload via a Word document attachment. Upon opening the attachment, the malicious code was designed to launch automatically. The BubbleBoy virus was the first to execute upon simply previewing the message – it was not necessary to open an attachment or to take any further action for the code to execute. BubbleBoy was developed using script embedded in the body of the e-mail message that executed as the message was rendered for viewing by the client. Finally, the recent ILOVEYOU worm was similar in concept to the Melissa virus in that it was transported as an e-mail attachment. In this case the attachment was not disguised as an innocuous Word document, but instead the attachment was a Visual Basic Script (.vbs) file that, upon launching, is interpreted and ran by the Windows Scripting Host (WSH).

The remainder of this document presents a variety of countermeasures that can be applied to limit the vulnerability of e-mail systems to these, and similar, attacks. It focuses primarily on the Microsoft Outlook clients, given the prominent role those applications played in recent incidents. Similar, the primary focus from an operating system perspective is Windows NT given it prevalent use in the Government and its vulnerability to such attacks. Inasmuch as the ILOVEYOU worm, and variants, are still topical at the time of this writing, Appendix A offer details regarding its impact upon a system.

Most of these recommendations are from a series of configuration guides written by NSA's Systems and Network Attack Center. Based upon an initial survey, it appears that the impact of the recent ILOVEYOU worm was greatly minimized in organizations that had followed these guidelines. For a complete set of our security configuration guides for Windows NT and commonly associated applications, call 1-800-688-6115. Request the "*Guide to Securing Microsoft Windows NT Networks and Applications*".

As always, test any of these procedures you may choose to implement on a test LAN prior to their usage on an operational network. While we have positive experiences with each of these recommendations, it is impossible for our testing to fully emulate other operational environments.

Countermeasures:

The following are a set of specific countermeasures that can help reduce the threat caused by the various kinds of malicious payloads effecting mail clients. Not only are these

countermeasures effective against the specific attacks listed above, but will help reduce, *but not eliminate,* the threat against other forms of e-mail based attacks.

Countermeasure 1 – Microsoft's E-mail Security Patches

As a direct result of the ILOVEYOU worm and other similar computer security incidences, Microsoft released a security patch for Outlook 98 and Outlook 2000. This patch improves the security of the clients by blocking file attachments that could contain malicious code. Attachments that present the greatest threat – referred to as "Level 1" attachments in the Microsoft lexicon -- are stripped from incoming messages and from all previously saved messages. The patch and a complete listing of the file types that are considered Level 1 are provided at http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnol/office/support/fixes/outcust.asp

This patch handles what is defined as "Level 2" attachments in a different manner. Level 2 attachments are not blocked, but instead the user is required to save them to the hard disk before executing. This is intended to cause the user to pause before acting and not just absent-mindedly launch a potentially malicious attachment. By default, no file types are included in Level 2; however, the administrator can define the files types that should be included in Level 2 as well as modify the file types defined as Level 1. There is a very notable caveat on the ability to modify the Level 1 and Level 2 definitions – this can only be done for users connecting to an Exchange server and who are not using .pst files for storage of mail messages[1]. This ability to modify the Level 1 and Level 2 definitions can be used to enforce local security polices. For example, one could use these features to add .doc files (Word documents) to the Level 1 file list.

The patch also controls programmatic access to the Outlook address book via the Outlook Object model and Collaborative Data Objects (CDO) as a countermeasure against malicious code that replicates by auto-forwarding itself to a user's contacts and provides protection against malicious embedded objects and scripts. A complete description and installation instructions are provided at the office update URL provided above.

Note that this patch only works with Outlook 98 and Outlook 2000 – there is no similar patch available for earlier versions of Outlook or Outlook Express.

---

[1] All users of Outlook 2000 can benefit from Office 2000 SR-1 which allows the definition of Level 2 file types (but not Level 1). Reference Microsoft Knowledge Base article Q259228 for details (http://support.microsoft.com/support/kb/articles/Q259/2/28.ASP). It is important to set the file attachment security settings within Outlook to "high" when using this patch (reference page 16). Another patch is available for to CDO access to the address book as well. It is available at http://office.microsoft.com/downloads/2000/cdo2k.aspx (Outlook 2000) or http://office.microsoft.com/Downloads/9798/Cdoup98.aspx (Outlook 98).

An enhanced version of this patch was released in August, 2001 which places further limits on Level 1 attachments. Details on the various versions of this e-mail security patch are available at http://support.microsoft.com/support/kb/articles/Q262/6/31.ASP.

Countermeasure 2 – Use of Internet Explorer Security Zones

Outlook 98/2000 and Outlook Express 4.0/5.0 clients can take advantage of Internet Explorer security zones to protect against malicious code (ActiveX controls, Java, or scripts) embedded into the body of messages. Internet Explorer includes a capability to restrict the execution of such code based upon four zones. Before jumping into how Outlook uses these settings, a quick review of their use in Internet Explorer is in order.

- Local Intranet zone: This zone contains addresses that are typically behind the organization's firewall or proxy server. The default security level for the Local Intranet zone is "medium-low".
- Trusted Sites zone: This zone contains sites that are trusted -- sites that are believed not to contain files that could corrupt the computer or its data. The default security level for the Trusted Sites zone is "low".
- Restricted Sites zone: This zone contains sites that are not trusted -- that is, sites that may contain content that, if downloaded or ran, could damage the computer or its data. The default security level for the Restricted Sites zone is "high".
- Internet zone: By default, this zone contains anything that is not on the computer or an intranet, or assigned to any other zone. The default security level for the Internet zone is "medium".

A plethora of security related settings can be configured for each of these zones. Microsoft has canned policies defined as *low*, *medium-low, medium*, and *high* which the user can select or alternately the user can tailor the settings to his or her specific needs.

Outlook utilizes these zones in that the user can select which of two zones -- the Internet zone or the Restricted zone -- Outlook messages fall into. The settings for the selected zone are then applied by Outlook to all messages.

It is recommended to select the Restricted zone. To do so, select Tools/Options and the *Security* tab. Select *Restricted sites* from the zone drop-down box.

Set the settings for the Restricted zone as recommended below by selecting *Zone Settings* and clicking on *Custom Level*. Note that changes made here will also apply to the Restricted zone when web surfing with Internet Explorer. These recommendations apply specifically to Internet Explorer 5.5; the options available under Internet Explorer 5.0 and 4.0 are similar but do not include all of the settings[2].

- Download signed ActiveX controls - DISABLE
- Download unsigned ActiveX controls - DISABLE
- Initialize and script ActiveX controls not marked as safe - DISABLE
- Run ActiveX controls and plug-ins - DISABLE
- Script ActiveX controls marked safe for scripting - DISABLE
- Allow cookies that are stored on your computer – DISABLE

[2] Note that the Outlook 98/Outlook 2000 e-mail security patch sets those clients to use the restricted sites zone. It does not, however, comply with the specific settings detailed here for the variety of security attributes attributable to that zone.

- Allow per-session cookies (not stored) - DISABLE
- File download - DISABLE
- Font download - DISABLE
- Java permissions – DISABLE JAVA
- Access data sources across domains – DISABLE
- Don't prompt for client certificate selection when no certificates or only one certificate exists -- DISABLE
- Drag and drop or copy and paste files - DISABLE
- Installation of desktop items - DISABLE
- Launching programs within an IFRAME – DISABLE
- Navigate sub-frames across different domains - DISABLE
- Software channel permissions - HIGH SAFETY
- Submit nonencrypted form data - DISABLE
- Userdata persistence - DISABLE
- Active scripting - DISABLE
- Allow paste operations via script - DISABLE
- Scripting of Java Applets - DISABLE
- Logon - Anonymous logon

Note that following these recommendations will disable many advanced features; however, for the vast majority of e-mail users there will be no operational impact. This is because most e-mail messages are simple text messages with attachments. The features that are disabled deal primarily with script and controls embedded within the body of the message which are not important to typical e-mail users.

Note once again that these settings are shared with the Internet Explorer browser and web pages typically DO incorporate the kinds of features which are disabled via these settings. While this could represent an operational impact, keep in mind that the Restricted zone is intended to include those sites that are not trusted - one should restrict what those sites can do and in fact these recommended settings are only slightly more restrictive than the default settings for this zone.

These settings will counter known attacks that use active content contained within the body of e-mail messages such as the BubbleBoy virus.

Countermeasure 3 – Changing File Associations or Disabling WSH

The e-mail security patch described in Countermeasure 1 will offer protection against the ILOVEYOU worm and similar kinds of executable content in Outlook 98 and Outlook 2000. Unfortunately, there is no similar patch available for Outlook Express. A level of protection can be achieved in Outlook Express environments by changing the default action associated with potentially dangerous file types. The ILOVEYOU worm is propagated as a Visual Basic Script file (.vbs) which, upon launch by an unwitting recipient, is interpreted by the Windows Scripting Host. An effective countermeasure against this kind of attack is to change the default action that occurs when a user launches

(e.g., double-clicks) the .vbs file. In Windows NT this is accomplished via Windows Explorer. Select View/Folder Options, select the *VBScript Script File* entry, click *Edit*, highlight *Edit* in the *Actions* window, and click *Set Default*. With these changes invoked, if a user launches a .vbs attachment it will not be executed by the Windows Scripting Host. Instead, it will harmlessly open in the default editor (typically Notepad).

This action should be completed not just for .vbs files, but also for all code types interpretable by the Windows Scripting Host. While the ILOVEYOU worm utilized a .vbs file, other types of code also offer viable options for an attacker. By default, the following file types can be executed by the Windows Scripting Host. Each should be changed such that the default action is *Edit*.

- WSC
- WSH
- WS
- WSF
- VBS
- VBE
- JS
- JSE

In addition, there are third party extensions available for the Windows Scripting Host which allows it to interpret other forms of code such as Perl or TCL. The default action for any third party extensions should be changed as well.

While this approach works well for Outlook Express environments, it is important to note that there is no guarantee that all e-mail clients will consult the default action setting when a user launches an attachment. For example, when opening a .vbs attachment under certain Netscape Messenger releases, the user is presented with a choice to either open or save the attachment. If the user selects *open*, the code will be executed regardless of the default action setting. A second option, which avoids this potential problem, is to disable the Windows Scripting Host. This is fairly easy to do: simply rename the core Windows programs that support script execution (wscript.exe and cscript.exe). On Windows NT systems, these files reside in the %systemroot%\system32 directory (typically c:\winnt\system32)[3]. It is best to do this from the command line or from a batch file. If the name is changed from Windows Explorer some versions of the Windows operating system will automatically update file associations to reflect the new name – which, of course, renders the change ineffective.

Finally, a third option for disabling the Windows Scripting Host is to change the file permissions on cscript.exe and wscript.exe. This may be the preferred option if it is

[3] Note that renaming the files in Windows 2000 is a little tricky due to the protection Windows 2000 provides core files. To rename the files in Windows 2000, first rename them in %SystemRoot%\system32\dllcache and then rename them in %SystemRoot%\system32. Cancel the "Windows File Protection" dialog box when it appears.

desired, for example, to allow administrators access while denying general users the ability to execute scripts.

It is important to note that while this countermeasure is effective against ILOVEYOU and similar threats, it cannot possibly eliminate all risk as there are other file types that could contain malicious code as well. A simple example to illustrate this point is .exe files – they are obviously critical to the operation of a PC and cannot be disabled, yet could easily be used as a malicious code delivery mechanism.

Countermeasure 4 – MS Office Macro Protection and User Education

Microsoft provides for protection against some of malicious file attachments through the associated application. For example, even though by default the e-mail security patch of Countermeasure 1 does not address malicious Word macros, the Microsoft Office 97 suite offers optional macro protection mechanisms that can help counter the threat by identifying files that contain macros and offering the user the ability to disable the macros prior to launching the file. The application is not making any value judgments in relation to the code – this is left to the user who must respond appropriately to the prompt. User education is paramount – users must understand the risk associated with any form of code received from untrusted sources and know how to act appropriately. To enable this feature, select Tools/Options/General and enable *Macro Virus Protection*.

Office 2000 and Office XP enhances this functionality in that it can be configured to only run macros that have been digitally signed by a trusted entity. In Word, PowerPoint, and Excel these options are assessable via Tools/Macro/Security. Select *High* for maximum protection.

Countermeasure 5 - Displaying File Extensions

A common technique used to disguise malicious code is to make an executable appear as an innocuous file type. One way of doing this is to simply name the file with a superfluous file extension such as:

ILOVEYOU.TXT.VBS

If Windows is not configured to display file extensions, then this file, when viewed from Windows Explorer, would appear as a simple text file as in:

ILOVEYOU.TXT

In order to preclude this kind of masquerading, two actions must be taken. First, set Windows to display file extensions via the Windows Explorer. Select Options/View and disable (clear the check box) *Hide file extensions for known file types*. Unfortunately, for certain file types that can contain or point to executable components this setting has no effect. To configure Windows to display these file extensions delete the following keys:

File Extension	Registry Key	Notes
.lnk	HKEY_CLASSES_ROOT\lnkfile\NeverShowExt	Shortcut
.pif	HKEY_CLASSES_ROOT\piffile\NeverShowExt	Program information file (shortcut to a DOS program)
.scf	HKEY_CLASSES_ROOT\SHCmdFile\NeverShowExt	Windows Explorer Command file
.shb	HKEY_CLASSES_ROOT\DocShortcut\NeverShowExt	Shortcut into a document
.shs	HKEY_CLASSES_ROOT\ShellScrap	Shell Scrap Object
.xnk	HKEY_CLASSES_ROOT\xnkfile\NeverShowExt	Shortcut to an Exchange folder
.url	HKEY_CLASSES_ROOT\InternetShortcut\NeverShowExt	Internet shortcut
.maw	HKEY_CLASSES_ROOT\Access.Shortcut.DataAccessPage.1\NeverShowExt	The remainder are a series of shortcuts to elements of an MS Access database. Most components of an Access database can contain an executable component.
.mag	HKEY_CLASSES_ROOT\Access.Shortcut.Diagram.1\NeverShowExt	
.maf	HKEY_CLASSES_ROOT\Access.Shortcut.Form.1\NeverShowExt	
.mam	HKEY_CLASSES_ROOT\Access.Shortcut.Macro.1\NeverShowExt	
.mad	HKEY_CLASSES_ROOT\Access.Shortcut.Module.1\NeverShowExt	
.maq	HKEY_CLASSES_ROOT\Access.Shortcut.Query.1\NeverShowExt	
.mar	HKEY_CLASSES_ROOT\Access.Shortcut.Report.1\NeverShowExt	
.mas	HKEY_CLASSES_ROOT\Access.Shortcut.StoredProcedure.1\NeverShowExt	
.mat	HKEY_CLASSES_ROOT\Access.Shortcut.Table.1\NeverShowExt	
.mav	HKEY_CLASSES_ROOT\Access.Shortcut.View.1\NeverShowExt	

Countermeasure 6 – Keeping Up-to-Date with Patches

Many Internet based attacks utilize known vulnerabilities. The BubbleBoy virus is a good case in point where the author took advantage of known Internet Explorer vulnerabilities. Microsoft had already issued a patch for these vulnerabilities which renders BubbleBoy ineffectual.

Countermeasure 7 – Anti-Virus Products

Most virus scanning products function based upon scans for known virus signatures; therefore, they are ineffective against new or uncharacterized attacks. However, they can be effective at preventing reoccurrences of past attacks. Some anti-virus products allow the blocking of mail attachments at the mail server – this may be of value in stopping an outbreak of attachment-based malicious code within an organization in the interim before an update to an antiviral scanning tool's signature file is available. Being able to block attachments would allow basic e-mail connectivity but preclude infection by viruses that use attachments as the transport media such as Melissa and the ILOVEYOU attacks.

Countermeasure 8 – Respecting the Concept of Least Privilege

Least privilege is a basic tenet of computer security that basically means "giving a user only those rights that s/he needs to do their job". Malicious code runs in the security context on which it was launched – practically speaking, this means in the context of the user launching the code. Good practices include making certain that administrative accounts are kept to a minimum, that administrators use a regular account as much as possible instead of logging in as administrator to do routine things such as reading their mail, and setting resource permissions properly.

Countermeasure 9 – Operating System Security

Protection against malicious code can be greatly improved by controlling access to key system components. There are three distinct concerns in this regard – protecting critical elements of the System Registry, eliminating a known privilege elevation attack, and restricting access to Windows NT system directories.

Countermeasure 9a – Securing the System Registry

The ILOVEYOU worm took advantage of the weak permissions on the RUN and RUNSERVICES registry keys (reference Appendix A for a more detailed description of the ILOVEYOU worm). Since the default access permissions associated with these keys allow a user to CREATE or MODIFY the contents, it was able to write to these keys and set the Trojan scripts to run every time someone logged on to the workstation or server. The Registry key permissions should be per the following recommendations. These recommendations include the specific keys used by ILOVEYOU plus some additional keys that pose the same opportunity for malicious code writers.

Registry Key	User Group	Permissions
\MACHINE\SOFTWARE\Microsoft\Windows *key and subkeys* Parameters used by the Win32 subsystem.	Administrators Authenticated Users CREATOR OWNER SYSTEM	Full Control Read, Write, Execute Full Control Full Control
\MACHINE\SOFTWARE\Microsoft\Windows\ CurrentVersion\Run *key and subkeys* Contains names of executables to be run each time the system is started.	Administrators Authenticated Users SYSTEM	Full Control Read, Execute Full Control
\MACHINE\SOFTWARE\Microsoft\Windows\ CurrentVersion\RunOnce *key and subkeys* Contains the name of a program to be executed the first time a user ever logs on.	Administrators Authenticated Users SYSTEM	Full Control Read, Execute Full Control
\MACHINE\SOFTWARE\Microsoft\Windows\ CurrentVersion\RunOnceEx *key and subkeys* Contains setup information for some system components and Internet Explorer. Works much the same way as the RunOnce key.	Administrators Authenticated Users SYSTEM	Full Control Read, Execute Full Control
\MACHINE\SOFTWARE\Microsoft\Windows\ CurrentVersion\Shell Extensions *key and subkeys* Contains all shell extension settings, which are used to extend and expand the Windows NT interface.	Administrators Authenticated Users CREATOR OWNER SYSTEM	Full Control Read, Execute Full Control Full Control

It is also recommended that the following setting be invoked to preclude remote access to the Windows NT registry. To restrict network access to the registry, create and/or set the following registry key and key value:

Hive: HKEY_LOCAL_MACHINE
Key: \System\CurrentControlSet\Control\SecurePipeServers\winreg
Name: RestrictGuestAccess
Type: REG_DWORD
Value: 1

Countermeasure 9b – Securing Additional Base Named Objects

Securing base objects prevents malicious code from gaining local administrator privileges by way of a dynamic-link library (DLL). Without this heightened security, malicious code could load into memory a file with the same name as a system DLL and redirect programs to it. Use the Registry Editor to create and set the value of the following registry key:

Hive: HKEY_LOCAL_MACHINE
Key: \System\CurrentControlSet\Control\Session Manager
Name: AdditionalBaseNamedObjectsProtectionMode
Type: REG_DWORD
Value: 1

Countermeasure 9c – Securing the System Directories

The ILOVEYOU worm also took advantage of the fact that users were allowed to write to the system directories (WINNT/SYSTEM32 and WINNT/SYSTEM). It is recommended that Authenticated Users only have *Read* permissions to these directories and files and should not be able to create or write to the system directories. The recommended settings are as follows:

FOLDER OR FILE	USER GROUPS	RECOMMENDED PERMISSIONS
%WINNT% *folder, subfolders, and files* Contains many operating system executable programs.	Administrators Authenticated Users CREATOR OWNER SYSTEM	Full Control Read, Execute Full Control Full Control
%WINNT/SYSTEM% *folder, subfolders, and files* Contains many operating system DLLs, drivers, and executable programs.	Administrators Authenticated Users CREATOR OWNER SYSTEM	Full Control Read, Execute Full Control Full Control
%WINNT/SYSTEM32% *folder, subfolders, and files* Contains many operating system DLLs, drivers, and executable programs (32 bit. Programs)	Administrators Authenticated Users CREATOR OWNER SYSTEM	Full Control Read, Execute Full Control Full Control

Automation

Overview

The instructions provided thus far in this document are based on manipulating settings via the GUI. This is, of course, problematic in anything but the smallest of networks. Fortunately, there is an alternative to directly manipulating the GUI in that the configuration settings can be applied by effecting changes in the registry. A brief overview on how to do so is offered.

There are 2 fairly easy methods for changing the registry in an automated fashion – either by use of a .reg file or an .ini file. A .reg file is created with the commonly used tool *regedit.exe*. The .reg file is created by exporting a registry key of interest. The resulting .reg file can be merged into the registry on another machine by simply double-clicking it in Windows Explorer or calling it from a .bat file. Running *regedit.exe* with the /s option will preclude the need for the user to clear message box popups each time a .reg file is

run. This is something that would cause many users consternation if the files were ran without their explicit knowledge – such as part of a logon script.

The 2nd method to update the registry is to use .ini files with the *regini.exe* program to execute registry changes. *Regini.exe* is a utility available on the Windows NT resource kit. Use of *regini.exe* avoids popup messages altogether, but requires specific tree-like structures within precise .ini files. Included with the NT resource kit is *regini.doc* which is a valuable aid in understanding the required tree structure and proper syntax necessary for use with *regini.exe*. For instance a .reg file uses this line:

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\3]

But .ini files typically look like this:

```
HKEY_CURRENT_USER\Software\Microsoft
        Windows
                CurrentVersion
                                Internet Settings
                                Zones
                                        =
                                        3
```

Similarly, .reg entries look like this:

```
"1001"=dword:00000003
```

while .ini files look like this:

```
1001 = REG_DWORD 0x00000003
```

Fortunately, there is a much easier method of creating .ini files other than manually editing a .reg file. That is, use the *regdmp.exe* utility which is also available in the Resource Kit. *Regdmp.exe* allows one to dump the registry to a text file then cut out the parts not required. *Regdmp.exe* can dump the entire registry or a specific registry path as in:

```
regdmp hkey_current_user\software > new.ini
```

Armed with these utilities, one could automate the application of these settings by:

- Determining which registry key(s) control the desired settings (most are listed below)
- Setting, via the GUI, the desired settings
- Exporting those keys with *regdmp.exe*
- Creating an appropriately structured .ini file from the resultant output
- Calling *regini.exe [path]\[filename]* from the domain login script

Now, with each domain login, the appropriate Internet Explorer and Outlook Express security settings are applied. Note that this approach only works for users logging into a domain as is typically the case in larger networks – users logging directly into their host would be unaffected.

Location of Some Relevant Registry Settings

- IE security settings
  HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones
- Outlook Zone settings
  - Outlook Express -
    HKEY_CURRENT_USER\Identities\[Identity]\Software\Microsoft\Outlook Express\5.0\Security Zone
  - Outlook 98 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\8.0\Outlook\Options\General\Security Zone
  - Outlook 2000 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Outlook\Options\General\Security Zone
- File attachment setting
  - Outlook Express – Not applicable
  - Outlook 98 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\8.0\Outlook\Options\General\AttachmentSafety
  - Outlook 2000 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Outlook\Options\General\AttachmentSafety
- Default actions for code using, as an example, VBS:
  HKEY_CLASSES_ROOT\VBSFile\Shell
- Office Macro Protection
  - Word 97 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\8.0\Word\Options\EnableMacroVirusProtection
  - Excel 97 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\8.0\Excel\Microsoft Excel\Options6
  - PowerPoint 97 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\8.0\PowerPoint\Options\MacroVirusProtection
  - Word 2000 -
    HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Word\Security\Level

- Excel 2000 – HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\Excel\Security\Level
- PowerPoint 2000 - HKEY_CURRENT_USER\Software\Microsoft\Office\9.0\PowerPoint\Security\Level

Any other registry keys of interest can be determined by using utilities which monitor registry access such as Regmon, available from System Internals at http://www.sysinternals.com/, and ConfigSafe, available from ImagineLAN at http://www.configsafe.com/.

Further Information

To obtain our complete set of security configuration guides for Windows NT and commonly associated applications, call 1-800-688-6115. Request the "*Guide to Securing Microsoft Windows NT Networks and Applications*". A similar set of guides for the Windows 2000 environment is available at http://www.nsa.gov.

Appendix A - Summary of the ILOVEYOU Worm Actions

The worm is contained in a Visual Basic Script mail attachment. It is activated when the user attempts to open the attached document.

The first thing ILOVEYOU does is copy itself to the system folder into files named MSKernel32.vbs and LOVE-LETTER-FOR-YOU.vbs. It then copies itself to the windows folder into the file Win32DLL.vbs.

After copying itself to these three locations, it modifies two registry keys HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Current Version\Run\MSKernel32 and HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Current Version\RunService\Win32DLL.

These keys will invoke the worm upon subsequent system reboots.

ILOVEYOU then looks to see if the system directory contains the file WinFAT32.exe. If this file exists, indicating Windows 95 or 98, the malicious code reassigns Internet Explorer's start page to reference the file WIN-BUGFIX.exe on www.skyinet.net. It obtains this file from either the angelcat, chu, or koichi directories on skyinet. The next time Explorer is started, the Start Page will reference the remote file. This will cause the file to be downloaded to the compromised machine where the user will be asked if he wants to run it. The Start Page key is located in HKCU\Software\Microsoft\Internet Explorer\Main\Start Page. We were unable to access www.skyinet.net to obtain a copy of this file. Speculation on the Internet indicates that this program may be an agent to collect passwords and mail them to a central site.

ILOVEYOU then checks to see if it has already infected a machine. It does this by looking in the download directory for WIN-BUGFIX.exe. If this file can be found, the malicious code sets a Run key to invoke WIN-BUGFIX.exe from the download directory on subsequent reboots. The telltale sign for this is the registry key HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\WIN-BUGFIX.

ILOVEYOU then creates a LOVE-LETTER-FOR-YOU.HTM file in the system directory. This HTM file appears to contain VBScript logic that corresponds to its .vbs version.

The malicious code then proliferates through e-mail to everyone listed in the compromised user's address book. For each person listed in the address book, a mail message is created and a copy of LOVE-LETTER-FOR-YOU.vbs is attached to the message before it is sent.

After mailing copies to all members in the Address List, ILOVEYOU iterates through all drives on the system. If the DriveType is a known type (fixed, remote, CDROM, RAMDISK), it recurses through the drive's subfolders looking for .vbs, .vbe, .sct, .hta, .jpg, or .jpeg files. It overwrites any files with these extensions, with copies of the malicious code.

If it finds a mp2 or mp3 file, it creates a copy of itself in a .vbs file within the directory. The name of the file corresponds to the directory's name with a .vbs extension.

If ILOVEYOU finds a file named mirc32.exe, mlink.exe, mirc.ini, script.ini, or mirc.hlp it assumes that the directory is an Internet Relay Chat startup directory and creates a script.ini file there. This script is executed the next time the IRC Chat Client is started. From quick analysis, it appears that this script sends the malicious code to any machine that establishes an IRC connection with the infected host.

Appendix B – Windows 95/98 Countermeasures

Some of the countermeasures outlined in this paper are not applicable to Windows 9x environments since those platforms do not support security concepts such as Access Control Lists, Administrator verses non-administrative accounts, and etc.

The following is a listing of the countermeasures that are applicable to the Windows 9x environment. Note that Windows Scripting Host is an optional add-on for Windows 95 and therefore may not exist on all installations. It is, on the other hand, part of the default installation of Windows 98.

- Countermeasure 1 – Microsoft's E-mail Security
- Countermeasure 2 – Use of Internet Explorer Security Zones
- Countermeasure 3 – Changing File Associations or Disabling WSH – except for the discussion on file permissions
- Countermeasure 4 – MS Office Macro Protection and User Education
- Countermeasure 5 - Displaying File Extensions
- Countermeasure 6 – Keeping Up-to-Date with Patches
- Countermeasure 7 – Anti-Virus Products

Changes

Version 1.1 –

- Added detail concerning the ability of the Microsoft Office 2000 suite to limit macro execution to those that have been signed by a trusted entity.
- Detailed how *regedit.exe* can be used with an undocumented */s* option to suppress message boxes.
- Added Appendix B which details which of the countermeasures recommended in the paper are applicable to the Windows 95/98 environment.

Version 2.0 --

- Added details regarding the e-mail security patch that Microsoft released for Outlook 98 and Outlook 2000 in response to the ILOVEYOU worm and similar threats.
- Pointed out that if one wishes to disable cscript.exe or wscript.exe by changing their names it is best to do so from the command prompt.

Version 2.1

- Added recommendation to enable the display of file extensions.

Version 2.2

- Added details concerning a feature of Office 2000 SR-1 that allows the definition of file types that cannot be directly executed from a mail message but must instead be saved to the file system prior to execution. Of note is that this feature works for users who utilize .pst files while similar countermeasures do not.

Version 2.3

- Added detail concerning a patch for protecting against malicious code using CDO in Outlook 98/2000.
- Updated URLs to reflect the current location of referenced material.

Version 2.4

- Added a recommendation to delete a registry key so that .pif file extensions will be displayed.
- Updated a URL to reflect a change in the location of referenced material.

Version 2.41

- Updated recommendations to include one additional security setting available under IE 5.5.

Version 2.42

- Added the warnings and trademark pages.

Version 2.5

- Expanded the list of file extensions which should always be displayed.
- Updated information about the e-mail security patch to include information about a recent revision to the patch.

Version 2.6

- Updated some URLs.