

# Skeletons in Microsoft's Closet

- Silently Fixed Vulnerabilities



**Andre Protas**  
**Steve Manzuik**



eEye Digital Security®

# Presentation Outline

## - Introductions / Outline

- That's this slide so we are done with that.

## - Non-Disclosure

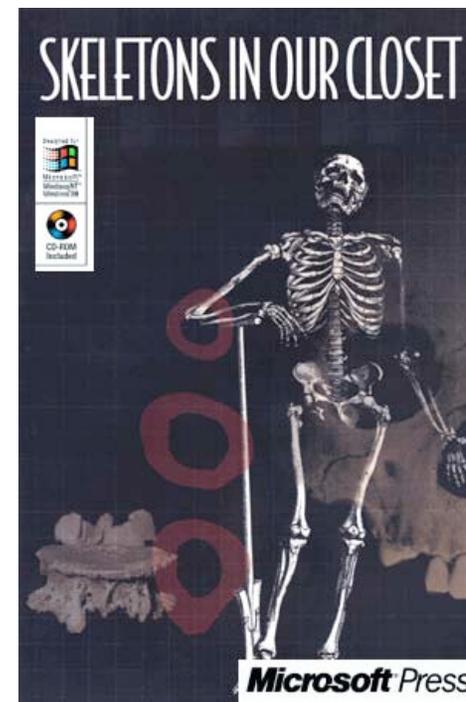
- Politics of not disclosing a vulnerability when it is fixed.
- What vendors practice this?
- Why is this bad?

## - Silently Fixed Bug Hunting

- Our methodology.
- Tools we used.
- Reversing 101.

## - Potential Hits

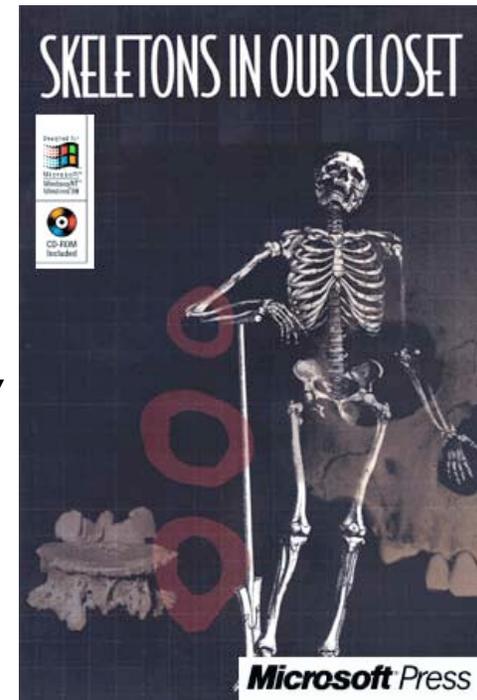
- Output from the tools.
- Identifying potential issues.



eEye Digital Security®

# Presentation Outline

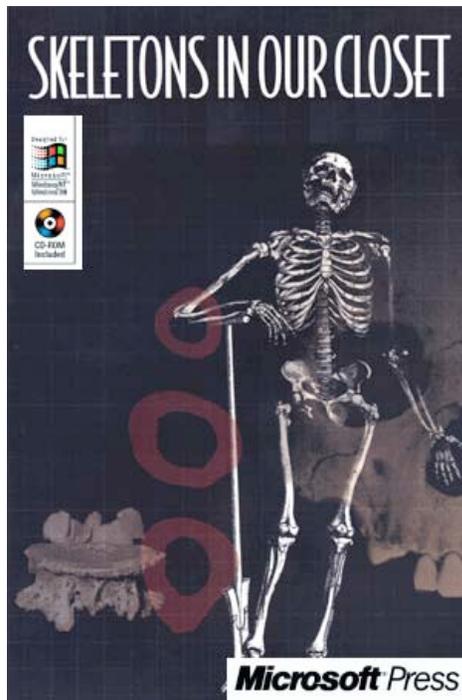
- Filtering the List
  - What we ignored.
  - Why we ignored?
- Vulnerabilities Found
  - Issues identified
  - Vulnerabilities vs. "Security Enhancements"
- Details
  - Security enhancements
  - Vulnerabilities found
- Vulnerability Exposed
  - Where we found the vulnerability
  - How do you exploit the vulnerability
- Demo
- Questions
  - **Buy us beer!!!**



## Why This Topic?

### ASN.1 Story

- Vendor released a patch (MS04-007) fixing what appeared to be 1 issue
- Vendor who discovered the vulnerability had 2 advisories
- Exploit code for a third issue was created and sold privately on the vulnerability market.



- Upon further analysis a total of **seven** issues were actually fixed.
- Apparently  $1 = 7$ . New math?
- So who cares? Or better yet why care?

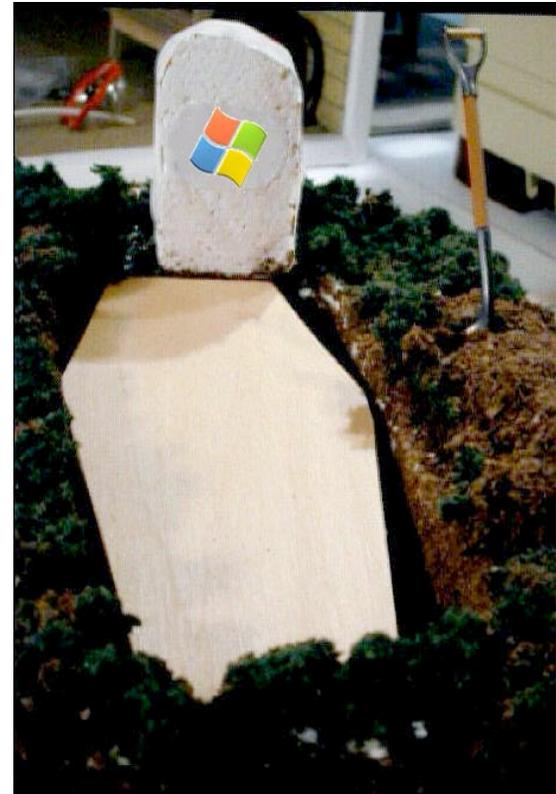
# NonDisclosure

## What is Non-Disclosure?

- In the context of Information Security Non-Disclosure is the act of not disclosing any details of a security vulnerability.
- Many vendors, not just Microsoft, practice this on a regular basis.

## Politics

- Disclosure = Press (bad?)
- Does press affect buying habits?
- Perception of security outweighs reality.



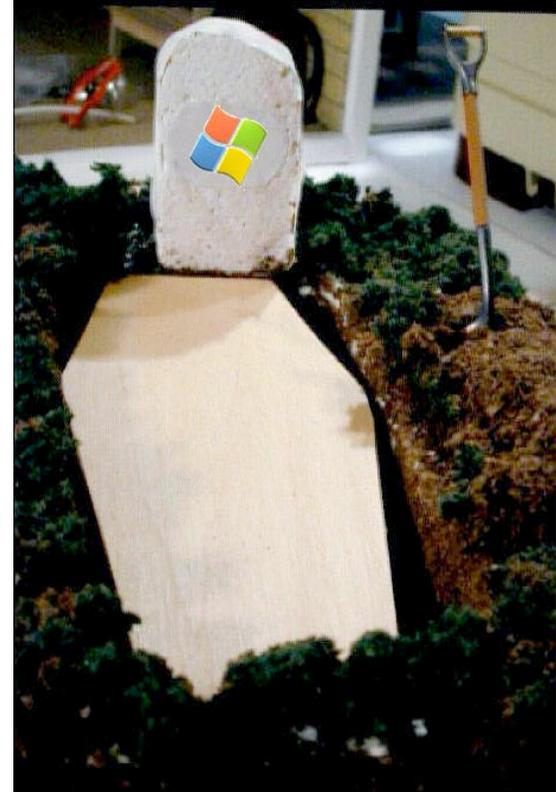
# NonDisclosure

## Vendor View

- Why disclosure internally found issues
- NDA agreements with third party consultants
- Adds release process overhead
- Customers install all patches anyways. Right?

*“If a vulnerability is found in a component, you should look for all related issues in that component”*

- Writing Secure Code Second Edition  
Microsoft Press



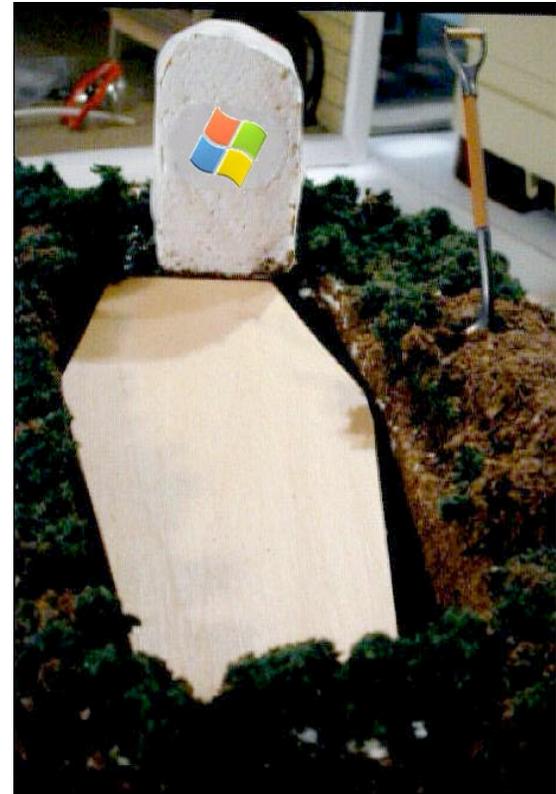
# NonDisclosure

## Why This Is Bad

- Customers do \*NOT\* install all patches.
- Affects patch management methodology.
- Signature based vendors may not catch on.

## A Word About Signatures

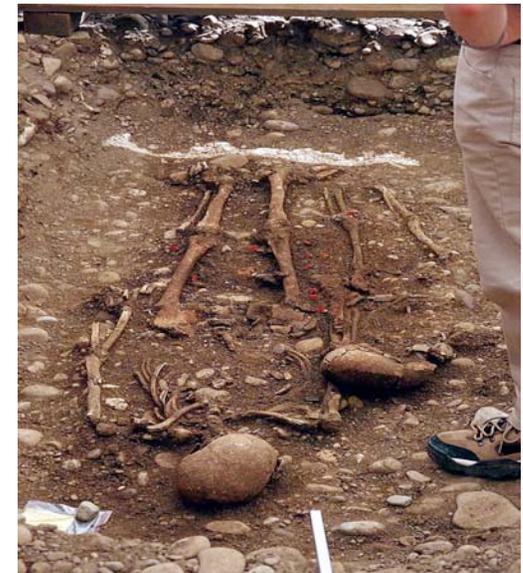
- Many vendors do not have the resources or skill to reverse a patch.
- The nature of our industry means that being first is best.
- Being first doesn't always equal being right.



# Silently Fixed Bug **Hunting**

## Methodology

- Identify patches that most likely have silent fixes
- Document publicly known issues addressed in the patch
- Catalog files in the patch
- Gather pre-patch files that are related
- Compare prepatch.dll with patch.dll
- Identify areas of interest
- Review interest areas for potentially exploitable flaws
- Exploit flaw in pre-patch environment
- Test exploit against post-patch environment
- Test against a signature based security solution



# Silently Fixed Bug Hunting

## Patch Identification

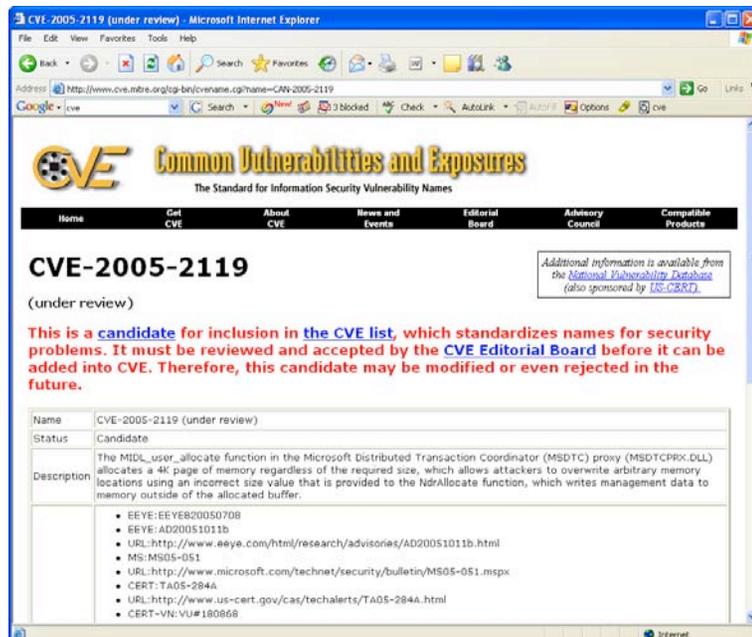
- Priority system for reviewing patches for silently fixed vulnerabilities.
  - Anonymous remotely accessible patched system functionality.
  - Non-Anonymous remote system functionality.
  - Non-remote system functionality



# Silently Fixed Bug Hunting

## Document publicly known issues addressed in the patch

- During Patch Tuesday, details may be at a minimum.
  - Use CVE and Advisory details (if they exist) to try to pinpoint the disclosed vulnerability.
  - Monitor exploit posts for the vulnerability to better understand the function that is being exploited.



The screenshot shows a Microsoft Internet Explorer browser window displaying the CVE-2005-2119 page. The page title is "CVE-2005-2119 (under review)". The main heading is "Common Vulnerabilities and Exposures" with the subtitle "The Standard for Information Security Vulnerability Names". The page content includes a navigation menu, a status section indicating the CVE is "under review", and a detailed description of the vulnerability. The description states: "The MIDL\_user\_allocate function in the Microsoft Distributed Transaction Coordinator (MSDTC) proxy (MSDTCPRX.DLL) allocates a 4K page of memory regardless of the required size, which allows attackers to overwrite arbitrary memory locations using an incorrect size value that is provided to the NdrAllocate function, which writes management data to memory outside of the allocated buffer." Below the description is a list of references including EYE:EEYE820050708, EYE:AD20051011b, and various URLs and CERT identifiers.

“The MIDL\_user\_allocate function in the Microsoft Distributed Transaction Coordinator (MSDTC) proxy (MSDTCPRX.DLL) allocates a 4K page of memory regardless of the required size, which allows attackers to overwrite arbitrary memory locations using an incorrect size value that is provided to the NdrAllocate function, which writes management data to memory outside of the allocated buffer. “

# Silently Fixed Bug Hunting

## Catalog Files In The Patch

- Usually pretty easy using the '/x' command on the installer.
- Filter update installer files out of the directory, and only include the files that were updated as part of the patch itself.

File Name	Version	Date	Time	Size
Catsrv.dll	2000.2.3529.0	05-Sep-2005	08:18	165,648
Catsrvut.dll	2000.2.3529.0	05-Sep-2005	08:18	595,728
Clbcatex.dll	2000.2.3529.0	05-Sep-2005	08:18	97,040
Clbcatq.dll	2000.2.3529.0	05-Sep-2005	08:18	551,184
Colbact.dll	2000.2.3529.0	05-Sep-2005	08:18	41,744
Comadmin.dll	2000.2.3529.0	05-Sep-2005	08:18	197,904
Comrepl.dll	2000.2.3529.0	05-Sep-2005	08:18	97,552
Comsetup.dll	2000.2.3421.3529	05-Sep-2005	08:18	342,288
Comsvcs.dll	2000.2.3529.0	05-Sep-2005	08:18	1,471,248
Comuid.dll	2000.2.3529.0	05-Sep-2005	08:18	625,936
Dtcsetup.exe	2000.2.3529.0	30-Aug-2005	04:47	1,833,968
Es.dll	2000.2.3529.0	05-Sep-2005	08:18	242,448
Msdctlog.dll	2000.2.3529.0	05-Sep-2005	08:18	96,016
Msdctprx.dll	2000.2.3529.0	05-Sep-2005	08:18	726,288
Msdcttm.dll	2000.2.3529.0	05-Sep-2005	08:18	1,200,400
Msdtcui.dll	2000.2.3529.0	05-Sep-2005	08:18	153,872
Mtstocom.exe	2000.2.3529.0	30-Aug-2005	05:05	155,408
Mtxclu.dll	2000.2.3529.0	05-Sep-2005	08:18	52,496
Mtxdm.dll	2000.2.3529.0	05-Sep-2005	08:18	26,896
Mtxlegih.dll	2000.2.3529.0	05-Sep-2005	08:18	35,600
Mtxoci.dll	2000.2.3529.0	05-Sep-2005	08:18	122,640
Ole32.dll	5.0.2195.7059	05-Sep-2005	08:18	957,712
Olecli32.dll	5.0.2195.7009	05-Sep-2005	08:18	69,392
Olecnv32.dll	5.0.2195.7059	05-Sep-2005	08:18	36,624
Rport4.dll	5.0.2195.6904	11-Mar-2004	21:29	449,808
Rpcss.dll	5.0.2195.7059	05-Sep-2005	08:18	212,240
Sp3res.dll	5.0.2195.7040	21-Apr-2005	10:07	6,309,376
Stclient.dll	2000.2.3529.0	05-Sep-2005	08:18	71,440
Txfaux.dll	2000.2.3529.0	05-Sep-2005	08:18	398,608
Xolehlp.dll	2000.2.3529.0	05-Sep-2005	08:18	19,216

# Silently Fixed Bug Hunting

## Gather pre-patch files that are related

- VMWare images can be a huge help here.
  - Keep VMWare images of each SP/UR as well as a current one for use against upcoming patches
- Keep a solid filing convention for the files that are to be analyzed to avoid confusing
  - Especially useful in batch analysis of service packs or update rollups



BEFORE



AFTER

# Silently Fixed Bug Hunting

## Diffing In General

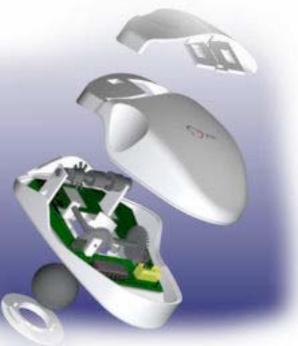
- The process of enumerating the changes made between two entities
  - Typically performed on files to look for textual differences (+ / - / change)
  - Great for learning the differences between configuration files
- But this can also be used in binary files.
  - Enumerate the functionality between two dlls/exe/etc files (+ / - / change)
  - Great for learning what security/functionality enhancements may have been introduced in the patch.
  - Use IDA Pro to reverse engineer the system file both pre- and post-patch.



# Silently Fixed Bug Hunting

## Reverse Engineering 101

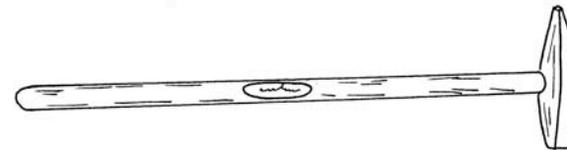
- A compiled file can be disassembled to show the machine code being processed for that file.
- Allows for pseudo-translation into source code.
- A disassembly be used to find flaws, hidden APIs, or any other number of low level functionality that may/may not be documented in standard references.
- Our Use? We use reverse engineering to dissect the security enhancements applied in Microsoft patches.

A screenshot of a disassembler showing C++ source code. The code is displayed in a window with a dark background and light text. The code includes class declarations and implementations for various classes, including CClientSocket, CRequest, and CClientSocket. The code is organized into several panes, showing different parts of the program's structure. The code is written in C++ and includes various methods and attributes.

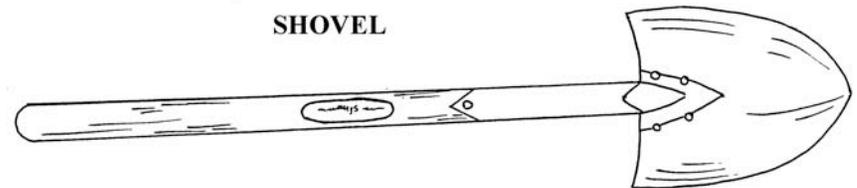
# Silently Fixed Bug Hunting

## Tools we used

- File information extraction
  - Muddle diff
  - Strings diff
  - Symbol retrieval
- Analysis with IDAPro
  - Custom IDA(Python)? Plugins
  - Sabre BinDiff
- Normal Debugging/Testing Environments
  - Pre/Post Patch VMWares (not `snapshots`)



MAUL



SHOVEL



# What About Service Packs / Update Rollups?

---

## Automation is key

- Enter BS (binary\_diffing starter)
- Useful suite of scripts to automate the basic binary diffing dependencies to allow for less wasted time.
  - Informational Gathering (sizes, names, versions, md5s, etc)
  - PDB symbol retrieval
  - IDB generation (pre/post PDB symbols)
  - Muddle/String diff utility
  - Allows for a specified IDC script to be run
- This tool allows for a complete basic reconnaissance of a service pack / update rollup once it has finished analyzing all of the files (pre and post patch) that were update.



# Potential Hits

## Output – IDA Split Screen

The image shows a split-screen view of the IDA Pro disassembler. The left pane displays assembly code for the address range 76891974 to 768919B4. The right pane displays assembly code for the address range 76892821 to 768928E7. Both panes show assembly instructions with their corresponding hex values and comments. The bottom of the image shows the IDA Pro interface with various toolbars and a status bar.

```
.text:76891974 push ebp
.text:76891975 mov ebp, esp
.text:76891977 sub esp, 40h
.text:76891978 push ebx
.text:76891979 push esi
.text:7689197A push edi
.text:7689197B mov edi, [ebp+arg_0]
.text:7689197C xor esi, esi
.text:7689197E push edi
.text:76891980 mov [ebp+var_4], esi
.text:76891982 call VerifyIfHandleOk
.text:76891984 cmp eax, esi
.text:76891986 jl loc_76891C49
.text:76891988 byte ptr [edi+10h], 2
.text:7689198A mov ebx, [ebp+arg_1C]
.text:7689198C push ebx
.text:7689198E call VerifyModuleStringOk
.text:76891990 cmp eax, esi
.text:76891992 jl loc_76891C40
.text:76891994 cmp [ebp+arg_24], esi
.text:76891996 jr loc_7689A007
.text:76891998
.text:76891998 loc_76891998:
.text:76891998 test byte ptr [edi+10h], 20h
.text:7689199A jnz loc_7689199E
.text:7689199C mov [ebp+var_24], esi
.text:7689199E short loc_76891998
.text:768919A0 push [ebp+var_24]
.text:768919A2 call ds:_imp_IsValidSid@4
.text:768919A4 test eax, eax
.text:768919A6 jz loc_768919E1
.text:768919A8
.text:768919A8 loc_768919A8:
.text:768919A8 word ptr [ebp+arg_14], 0
.text:768919AA cmp short loc_7689199E
.text:768919AC
.text:768919AC loc_768919AC:
.text:768919AC mov ecx, [ebp+arg_24]
.text:768919AE movzx eax, si
.text:768919B0 push dword ptr [ecx+eax*4]
.text:768919B2 call VerifyModuleStringOk
.text:768919B4 mov eax, eax
.text:768919B6 inc esi
.text:768919B8 cmp word ptr [ebp+arg_14]
.text:768919BA jb short loc_7689199E
.text:768919BC
.text:768919BC loc_768919BC:
.text:768919BC test byte ptr [edi+0Ch], 2
.text:768919BE jnz loc_7689199E
.text:768919C0 xor eax, eax
.text:768919C2 cmp [ebp+arg_28], eax
.text:768919C4 inc short loc_768919B8
.text:768919C6 jmp [ebp+arg_30], eax
.text:768919C8
.text:768919C8 loc_768919C8:
.text:768919C8 mov word ptr [ebp+var_4C], ax
.text:768919CA mov [edi+14h]
.text:768919CC lea esi, [edi+30h]
.text:768919CE lea ecx, [ebp+var_34]
.text:768919D0 push ecx
.text:768919D2 mov [ebp+var_20], esi
.text:768919D4 mov [ebp+var_28], ecx
.text:768919D6 call FindModuleStructFromToken
.text:768919D8 mov eax, eax
.text:768919DA mov [ebp+var_8], ecx
```



# Potential Hits

## Output – Strings Diff

```
REMOVED IN THE LATEST BUILD
```

```
-----  
_NDRSERVERCALL2  
_ELFRDEREGISTEREVENTSOURCE  
_IELF_HANDLE_RUNDOWN  
_ELFRCLOSEEL  
_FIXCONTEXTHANDLESFORRECORD  
-----
```

```
END OF OLD REMOVED
```

```
ADDED IN THE LATEST BUILD
```

```
-----  
_IMP_I_RPCBINDINGISCLIENTLOCAL  
_STRINGCOPYWORKERW  
_FIXCONTEXTHANDLESFORRECORD  
_PFNI_RPCSESSIONSTRICTCONTEXTHANDLE  
-----
```

```
END OF ADDED
```



# Potential Hits

## Output – BinDiff

Cha...	Function 1 EA	Function 1 Name	Function 2 EA	Function 2 Name
<b>B</b> No	76892491	_ElfRegistryMonitor@8	76891cde	_ElfRegistryMonitor@8
<b>B</b> No	76892455	_InitNotify@4	76891ca9	_InitNotify@4
<b>B</b> No	7689240d	sub_7689240D	76891c6a	_ElfSetupMonitor@4
<b>B</b> No	768923a9	sub_768923A9	768938dc	_ElfOpenELA@28
<b>B</b> No	7689a825	__local_unwind2	7689a0e5	__local_unwind2
<b>B</b> No	76891b00	sub_76891B00	76891435	_FindModuleStrucFromAtom@4
<b>B</b> No	76891b73	_ElfFreeBuffer@4	7689141c	_ElfFreeBuffer@4
<b>B</b> No	76891b84	sub_76891B84	7689155f	_VerifyElfHandle@4
<b>B</b> No	76891c0b	_ElfAllocateBuffer@4	76891401	_ElfAllocateBuffer@4
<b>B</b> No	76891c1c	_MIDL_user_allocate@4	7689140d	_MIDL_user_allocate@4
<b>B</b> No	76891c30	sub_76891C30	768914fd	_VerifyUnicodeString@4
<b>B</b> No	76891c9b	_ElfCloseAudit@8	768921c9	_ElfCloseAudit@8
<b>B</b> No	76891db5	sub_76891DB5	76891d29	_GetModuleStruc@4
<b>B</b> No	76891e15	sub_76891E15	76891fb2	_ElfGetPrivilege@8
<b>B</b> No	76892097	sub_76892097	76891e92	_ElfAccessCheckAndAudit@32
<b>B</b> No	768921be	sub_768921BE	768920c1	_ElfReleasePrivilege@0
<b>B</b> No	76892216	sub_76892216	768913e1	_GetELState@0
<b>B</b> No	7689223d	sub_7689223D	76891485	_ElfPerformRequest@4
<b>B</b> No	768922ea	sub_768922EA	7689387f	_VerifyAnsiString@4
<b>B</b> No	76892350	_ElfRegisterEventSourceA@24	7689382b	_ElfRegisterEventSourceA@24
<b>B</b> Yes	76891f2b	sub_76891F2B	76891d9d	_ElfOpenELW@28
<b>B</b> Yes	76892560	sub_76892560	768915e4	_PerformWriteRequest@4
<b>B</b> Yes	76892c7d	sub_76892C7D	768931b8	_SetupModules@12
<b>B</b> Yes	76893da5	sub_76893DA5	76893093	_ElfCreateLogFileObject@12
<b>B</b> Yes	76893f1a	SvcEntry_Eventlog	76892238	_SvcEntry_Eventlog@16
<b>B</b> Yes	76895021	sub_76895021	768938ae	_ReadFromLog@4
<b>B</b> Yes	76898bb8	sub_76898BB8	76897752	_FixContextHandlesForRecord@8
<b>B</b> Yes	7689a74e	sub_7689A74E	7689a02a	_ElfCleanUp@4

Line 95 of 135



# Filtering the List

---

## Ignored Results

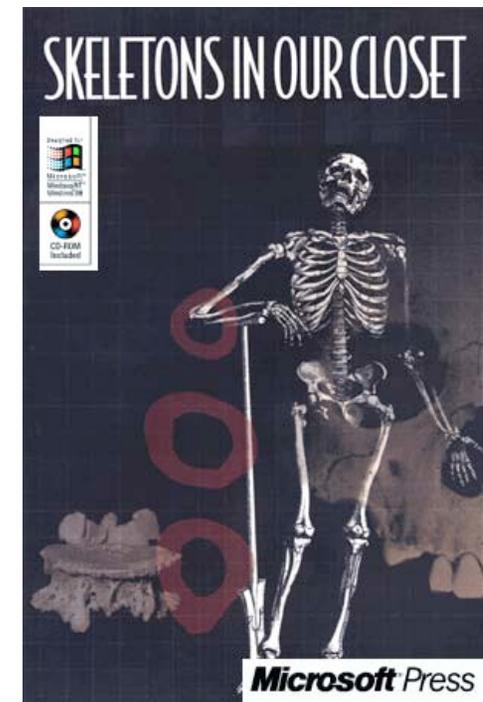
- Although we didn't ignore tool output completely, some information was not used as much except for a support role to the IDA twin-disassembly or the BinDiff IDA plugin.
- Strings\_Diff
- Removed subroutines from disassembly.
- Muddle\_Diff



# Filtering the List

## Why we ignored

- We ignored much of the string diff generation, as there were many false positives that were reported by the string generation tool.
- Although removed functionality could be interesting as well, we were primarily concerned with the added functionality / security enhancements.
- Most of the muddle output was ignored as muddle can generate many false positives (complicated data structures could be equal, but would alert a diffing tool as changed).



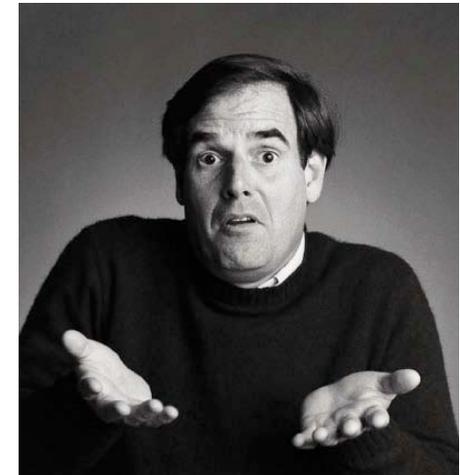
# Security Enhancements Found

## Not Vulnerabilities But...

- The release notes from Windows 2000 Update Rollup 1 says;

“This update rollup contains security-related updates that were produced for Windows 2000 between the release of Windows 2000 SP4 and April 30, 2005. On April 30, 2005, the contents of Update Rollup 1 were locked for final testing by Microsoft and customer beta testing. This update rollup also contains several important non-security updates. This article contains detailed information about this update rollup, answers frequently asked questions, and lists the updates that are included in this update rollup. ”

- Do you understand this paragraph?



# Security Enhancements Found

## Not Vulnerabilities But...

- Non-Strict RPC connections now enforced
  - Previously allowed for context switching between RPC interfaces within the same process (i.e. services.exe)
  - Allows for RPC evasion (via ALTER\_CONTEXT)
  - Potential DoS (access violation) from improperly checked context handle from possibly



Example: Eventlog.dll

CLSID: 82273fdc-e32a-18c3-3f78-827929dc23ea

NOTE: This is the ONLY dll with this change.

## Vulnerabilities Found

Don't worry. We were not done looking for silently fixed bugs.....

REMOVED IN THE LATEST BUILD

-----  
\_NDRSERVERCALL2  
\_ELFRDEREGISTEREVENTSOURCE  
\_IELF\_HANDLE\_RUNDOWN  
\_ELFRCLOSEEL  
\_FIXCONTEXTHANDLESFORRECORD  
-----

END OF OLD REMOVED

ADDED IN THE LATEST BUILD

-----  
\_IMP\_I\_RPCBINDINGISCLIENTLOCAL  
\_STRINGCOPYWORKERW  
\_FIXCONTEXTHANDLESFORRECORD  
\_PFNI\_RPCSESSIONSTRICTCONTEXTHANDLE  
-----

END OF ADDED

File Name	Version	Date	Time	Size
Catsrv.dll	2000.2.3529.0	05-Sep-2005	08:18	165,648
Catsrvut.dll	2000.2.3529.0	05-Sep-2005	08:18	595,728
Clbcatex.dll	2000.2.3529.0	05-Sep-2005	08:18	97,040
Clbcatq.dll	2000.2.3529.0	05-Sep-2005	08:18	551,184
Colbact.dll	2000.2.3529.0	05-Sep-2005	08:18	41,744
Comadmin.dll	2000.2.3529.0	05-Sep-2005	08:18	197,904
Comrepl.dll	2000.2.3529.0	05-Sep-2005	08:18	97,552
Comsetup.dll	2000.2.3421.3529	05-Sep-2005	08:18	342,288
Comsvcs.dll	2000.2.3529.0	05-Sep-2005	08:18	1,471,248
Comuid.dll	2000.2.3529.0	05-Sep-2005	08:18	625,936
Dtsetup.exe	2000.2.3529.0	30-Aug-2005	04:47	1,833,968
Es.dll	2000.2.3529.0	05-Sep-2005	08:18	242,448
Msdctlog.dll	2000.2.3529.0	05-Sep-2005	08:18	96,016
Msdctprx.dll	2000.2.3529.0	05-Sep-2005	08:18	726,288
Msdctm.dll	2000.2.3529.0	05-Sep-2005	08:18	1,200,400
Msdctui.dll	2000.2.3529.0	05-Sep-2005	08:18	153,872
Mtstocom.exe	2000.2.3529.0	30-Aug-2005	05:05	155,408
Mtxclu.dll	2000.2.3529.0	05-Sep-2005	08:18	52,496
Mtxdm.dll	2000.2.3529.0	05-Sep-2005	08:18	26,896
Mtxlegih.dll	2000.2.3529.0	05-Sep-2005	08:18	35,600
Mtxoci.dll	2000.2.3529.0	05-Sep-2005	08:18	122,640
Ole32.dll	5.0.2195.7059	05-Sep-2005	08:18	957,712
Olecli32.dll	5.0.2195.7009	05-Sep-2005	08:18	69,392
Olecnv32.dll	5.0.2195.7059	05-Sep-2005	08:18	36,624
Rport4.dll	5.0.2195.6904	11-Mar-2004	21:29	449,808
Rpss.dll	5.0.2195.7059	05-Sep-2005	08:18	212,240
Sp3res.dll	5.0.2195.7040	21-Apr-2005	10:07	6,309,376
Stolient.dll	2000.2.3529.0	05-Sep-2005	08:18	71,440
Txfaux.dll	2000.2.3529.0	05-Sep-2005	08:18	398,608
Xolehlp.dll	2000.2.3529.0	05-Sep-2005	08:18	19,216



# Vulnerabilities Found

## MS05-051 – MSDTC Vulnerability – CAN-2005-2119

- After being disappointed with Windows 2000 Update Release 1
- MS05-051 fixed only 1 MSDTC vulnerability and a few others
- But what is "wcscpy(arg\_28, pwszNULL\_GUID)"

Microsoft Security Bulletin MS05-051: Vulnerabilities in MSDTC and COM - Could Allow Remote Code Execution (902400)

Published: October 11, 2005 | Updated: October 25, 2005

**Version:** 1.2

**Summary**

**Who should read this document:** Customers who use Microsoft Windows

**Impact of Vulnerability:** Remote Code Execution

**Maximum Severity Rating:** Critical

**Recommendation:** Customers should apply the update immediately.

**Security Update Replacement:** This bulletin replaces several prior security updates. See the frequently asked questions (FAQ) section of this bulletin for the complete list.

**Caveats:** Systems that have been modified to non default Access Control List settings on the registration folder in the Windows directory may experience unexpected behavior. For more information, see [Microsoft Knowledge Base Article 902400](#).

**Affected Software and Security Update Download Locations:**

**Affected Software:**

- Microsoft Windows 2000 Service Pack 4 - [Download the update](#)
- Microsoft Windows XP Service Pack 1 and Microsoft Windows XP Service Pack 2 - [Download the update](#)
- Microsoft Windows XP Professional x64 Edition - [Download the update](#)
- Microsoft Windows Server 2003 and Microsoft Windows Server 2003 Service Pack 1 - [Download the update](#)
- Microsoft Windows Server 2003 for Itanium-based Systems and Microsoft Windows Server 2003 with SP1 for Itanium-based Systems - [Download the update](#)
- Microsoft Windows Server 2003 x64 Edition - [Download the update](#)

**Non-Affected Software:**

- Microsoft Windows 98, Microsoft Windows 98 Second Edition (SE), and Microsoft Windows Millennium Edition (ME)

The software in this list has been tested to determine whether the versions are affected. Other versions either no longer include security update support or may not be affected. To determine the support life cycle for your product and version, visit the [Microsoft Support Lifecycle Web site](#).

[Top of section](#)

**General Information**

- Executive Summary
- Frequently asked questions (FAQ) related to this security update
- Vulnerability Details
  - MS05-051: A remote code execution and local elevation of privilege vulnerability exists in the Microsoft Distributed Transaction Coordinator that could allow an attacker who successfully exploited this vulnerability to take complete control of the affected system.**
  - Mitigating Factors for MSDTC Vulnerability - CAN-2005-2119:**
  - Workarounds for MSDTC Vulnerability - CAN-2005-2119:**
  - FAQ for MSDTC Vulnerability - CAN-2005-2119:**

"A remote code execution and local elevation of privilege vulnerability exists in the Microsoft Distributed Transaction Coordinator that could allow an attacker who successfully exploited this vulnerability to take complete control of the affected system. "



## Security Flaws Found

---

### MSDTC (MS05-051)

- Heap Overflow: `CRpcIoManagerServer::BuildContext`
- Lack of input validation allows for overwrite of the 'pszGuidOut' argument with a null GUID string
- Attacks XP/2000 (BuildContextW opnum 7) **as well as** NT40 (BuildContext opnum 1).
- Interesting new string added: 'At least one of the buffers passed into BuildContext has an incorrect length.' (0x6DFDE24B)
- 4 new string length checks added

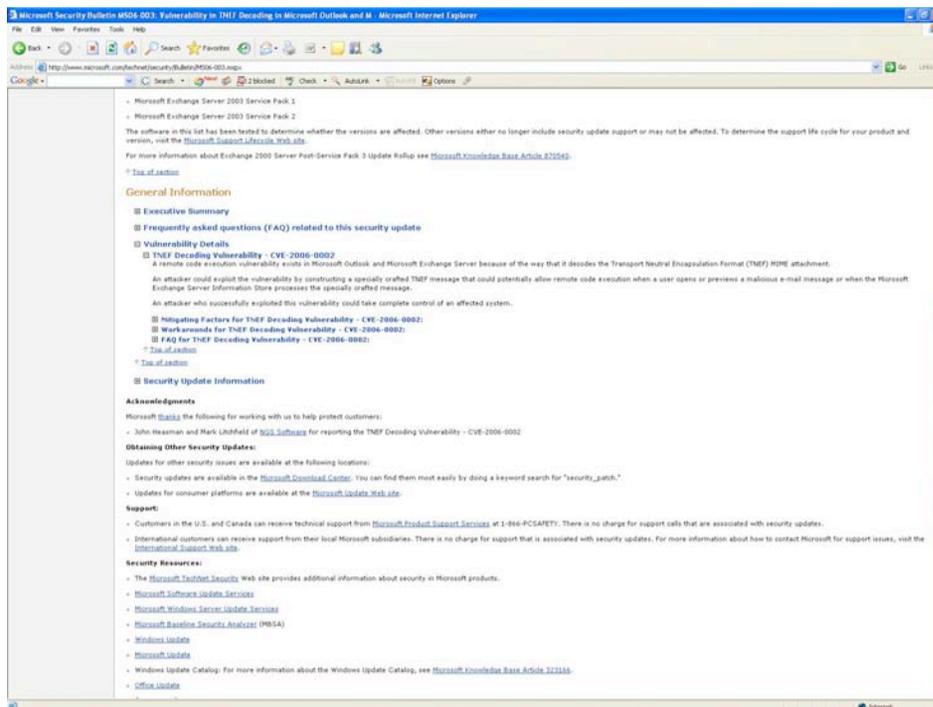


Found by Derek Soeder in a standard patch diffing session

# Vulnerabilities Found

## MS06-003 – Microsoft Exchange TNEF Issue

- After being disappointed with Windows 2000 Update Release 1
- Found multiple vulnerable functions
- But only 1 was reported in the advisory



### **"TNEF Decoding Vulnerability - CVE-2006-0002"**

*A remote code execution vulnerability exists in Microsoft Outlook and Microsoft Exchange Server because of the way that it decodes the Transport Neutral Encapsulation Format (TNEF) MIME attachment.*

*An attacker could exploit the vulnerability by constructing a specially crafted TNEF message that could potentially allow remote code execution when a user opens or previews a malicious e-mail message or when the Microsoft Exchange Server Information Store processes the specially crafted message.*

*An attacker who successfully exploited this vulnerability could take complete control of an affected system."*



eEye Digital Security®

## Security Flaws Found

---

### TNEF (MS06-003)

- Vulnerability reported from MS Security team only mentions `HrDecodeEncapsulation` .
- Many other changes were released in the patch within different functions.
- Example: `HrDecodeRecipTable`
- `new > 10000 (2710h)` check after `_WSTRM_Read` call
- Potentially exploitable (demo)
- Also added (encoding) updates to not allow malformed outbound TNEF.



NOTE: This is one of MANY size/length checks added in MS06-003.

# Demonstration

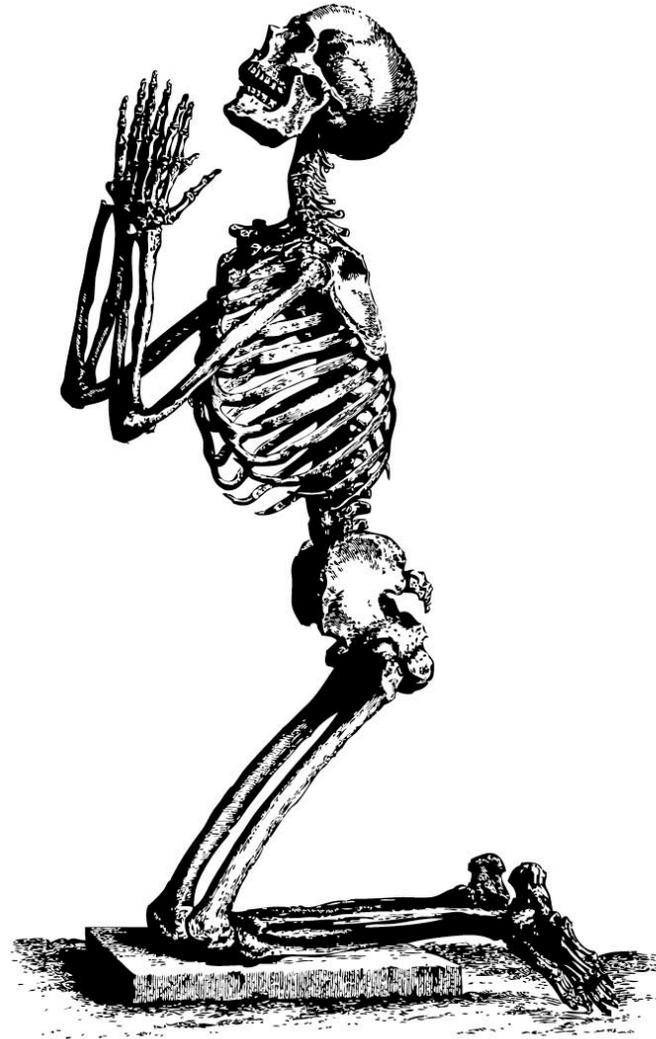
## Demo



# Demonstration

---

Demo time.....



## In Closing

---

- What you don't know can hurt you.
- Relying only on signatures can hurt you.
- Full-Disclosure from vendors would help.
- This is not just a Microsoft issue.**
  - Oracle
  - Apple
  - HP
  - IBM
  - **Other (Linux?)**



## References

---

- OpenRCE.org – Reverse Engineering Community
- Sabre-Security – Professional binary tools
- IDAPython – Python interface to IDA plugin API
- IDA Palace – Random IDA goodness
- eEye Blink – Generic Endpoint Security
- **Thanks: Derek Yoda Soeder, Barnaby “The Claw” Jack, Hugo The Puto**

# Questions

---



eEye Digital Security®