Binary Analysis: Finding Secrets in Code

Nish Bhalla
Founder, Security Compass

Agenda

Product Review Methodology

Basics

Debugging Steps

Ollydbg

Process Isolation

isDebuggerPresent

ISAPI Analysis

Typical Product Review Methodology

- **What it is:** Product reviews examine vendor applications using a black-box approach to find security vulnerabilities

- **Why:** Product reviews discover vulnerabilities before the product is shipped, thereby decreasing potential security updates, enhancing your organization's reputation amongst customers and creating a competitive advantage

- **How:** Product reviews examine applications to determine security vulnerabilities by performing the following steps:

1. Threat Analysis- Incorporate the full threat analysis methodology
2. Design Analysis- Map out the application use and functionality requirements
3. Architecture Review- Examine the dataflow model and the controls "by design" on each communication point to secure data
4. Application Penetration Testing- Perform black and grey-box testing on the product on a variety of potential end-user environments

Threat Analysis

- What is Threat Analysis?

- Threat Analysis or threat modeling is the process of systematically deriving the **key threats** relevant to an application in order to efficiently **identify** and **mitigate** potential **security weaknesses** before deployment

- It is a method to determine the unique threats that an application might face; it is a systematic method of finding security issues in an application by **forcing developers to think like an attacker**

- Security staff can **focus** their resources on the most important issues an application faces after performing this activity

Design And Architecture Review

- Design Review: Understand what the thought process was when the application was originally designed
- More often than not the design and the end product are two completely different applications
- Architecture Review: Evaluate the deployed application's architecture and implementation in a real world scenario

??? SECURE BY DEFAULT ???

Application Penetration Testing

- **Our Sample Application** is an ISAPI. It requires a pass phrase to browse into the website

- Why did I choose an ISAPI and not a stand alone application?

  - Developers often store credentials of back end databases in ISAPIs
  - ISAPIs are commonly where algorithms are stored
  - ISAPIs are typically not expected to land in the hands of a hacker
  - ISAPIs are typically reviewed for lost code and lost developers
  - ISAPIs can help you crash systems that are otherwise secure
  - I recently reviewed a few ISAPIs and I am partially basing this example on observations made in those reviews

Agenda

Product Testing Methodology

Basics

Debugging Steps

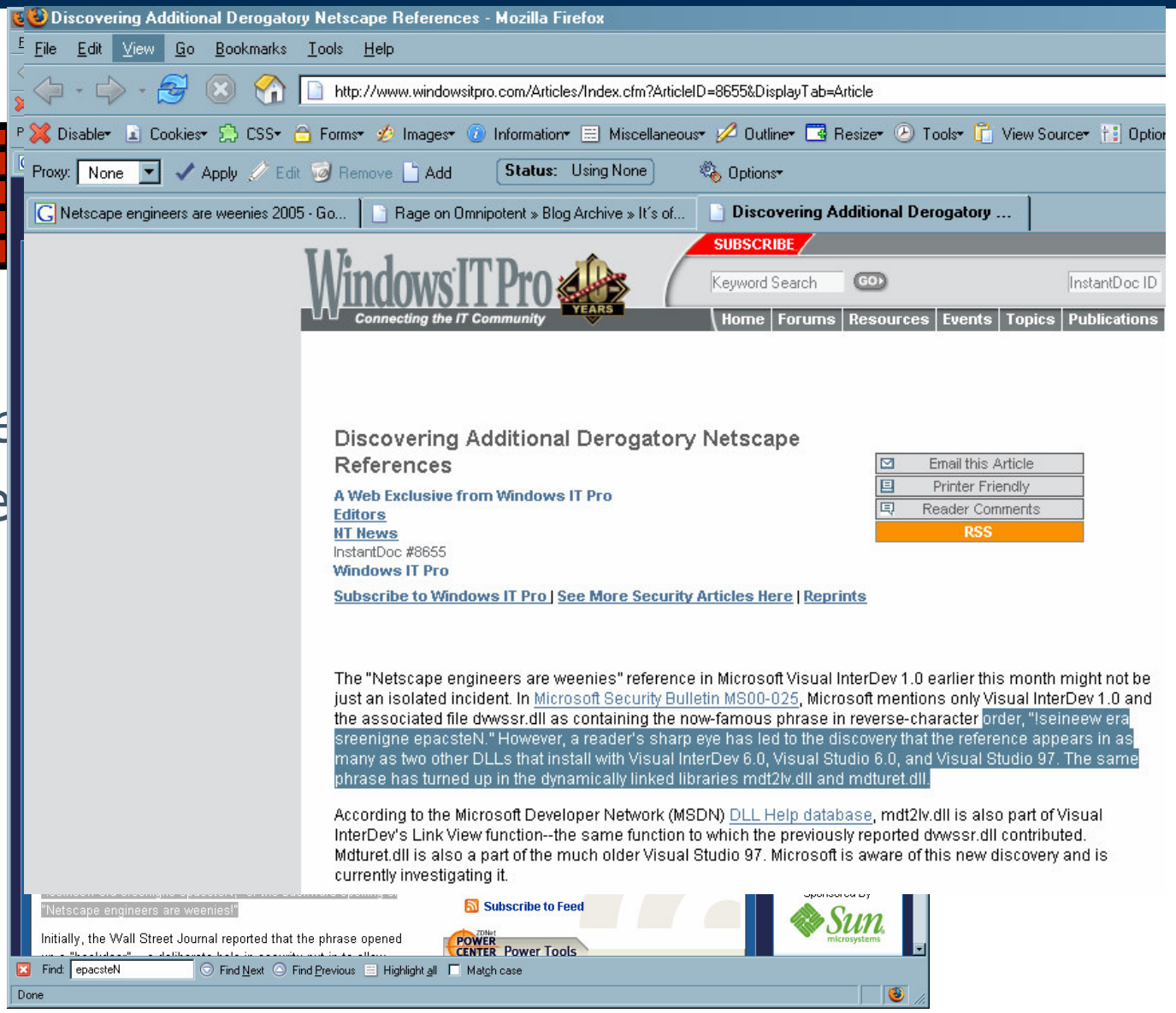
Ollydbg

Process Isolation

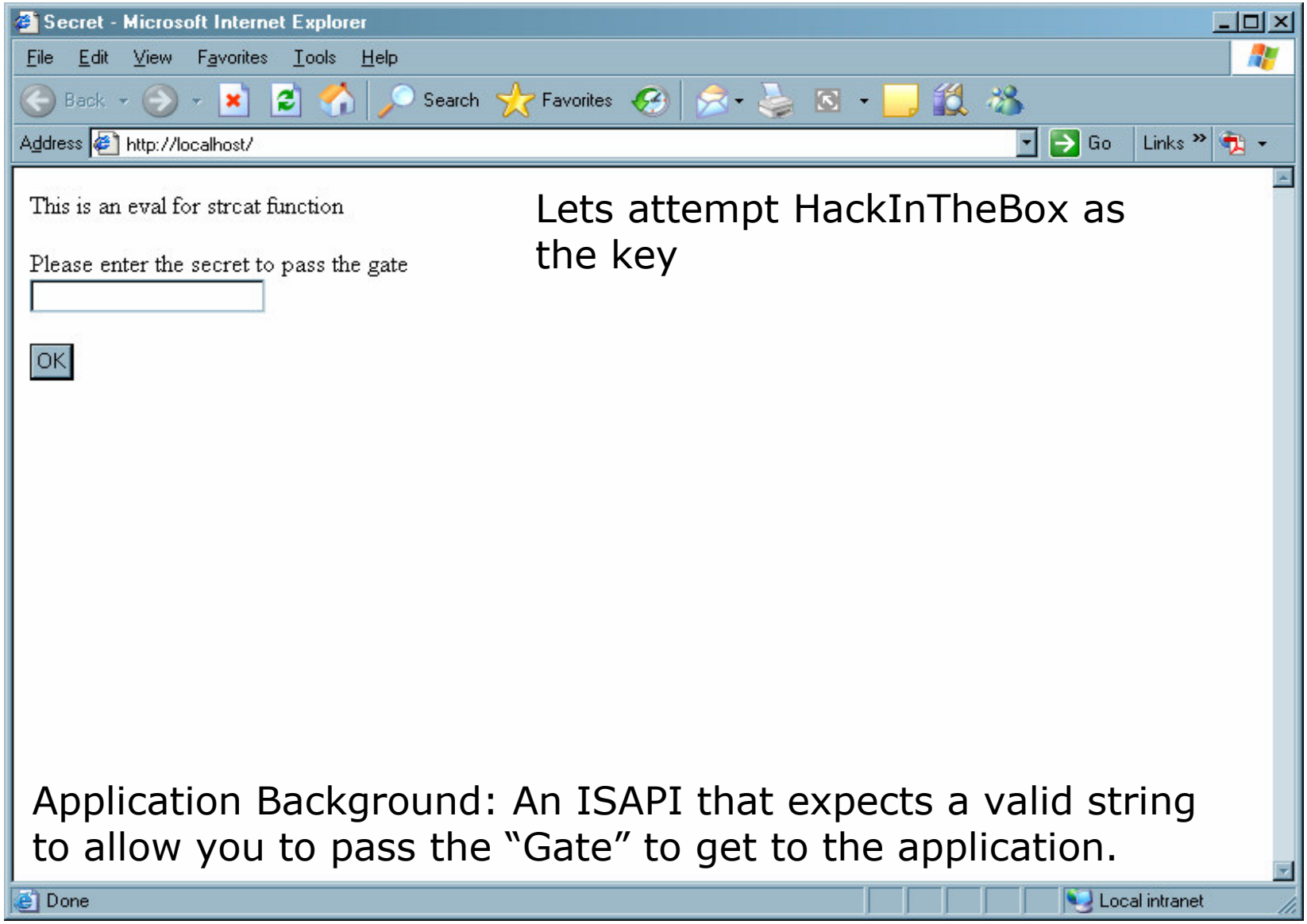
isDebuggerPresent

ISAPI Analysis

ISAPI


Discovering Additional Derogatory Netscape References - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.windowsitpro.com/Articles/Index.cfm?ArticleID=8655&DisplayTab=Article
Netscape engineers are weenies 2005 - Go...
Rage on Omnipotent » Blog Archive » It´s of...
Discovering Additional Derogatory ...
SUBSCRIBE
Windows IT Pro
Connecting the IT Community
Keyword Search
InstantDoc ID
Home Forums Resources Events Topics Publications
Discovering Additional Derogatory Netscape References
A Web Exclusive from Windows IT Pro
Editors
NT News
InstantDoc #8655
Windows IT Pro
Subscribe to Windows IT Pro | See More Security Articles Here | Reprints
Email this Article
Printer Friendly
Reader Comments
RSS
The "Netscape engineers are weenies" reference in Microsoft Visual InterDev 1.0 earlier this month might not be just an isolated incident. In Microsoft Security Bulletin MS00-025, Microsoft mentions only Visual InterDev 1.0 and the associated file dwwssr.dll as containing the now-famous phrase in reverse-character order, "Iseineew era sreenigne epacsteN." However, a reader's sharp eye has led to the discovery that the reference appears in as many as two other DLLs that install with Visual InterDev 6.0, Visual Studio 6.0, and Visual Studio 97. The same phrase has turned up in the dynamically linked libraries mdt2lv.dll and mdturet.dll.
According to the Microsoft Developer Network (MSDN) DLL Help database, mdt2lv.dll is also part of Visual InterDev's Link View function--the same function to which the previously reported dwwssr.dll contributed. Mdturet.dll is also a part of the much older Visual Studio 97. Microsoft is aware of this new discovery and is currently investigating it.
"Netscape engineers are weenies!"
Initially, the Wall Street Journal reported that the phrase opened
Subscribe to Feed
Power Tools
Sun microsystems
Find: epacsteN
Find Next
Find Previous
Highlight all
Match case
Done

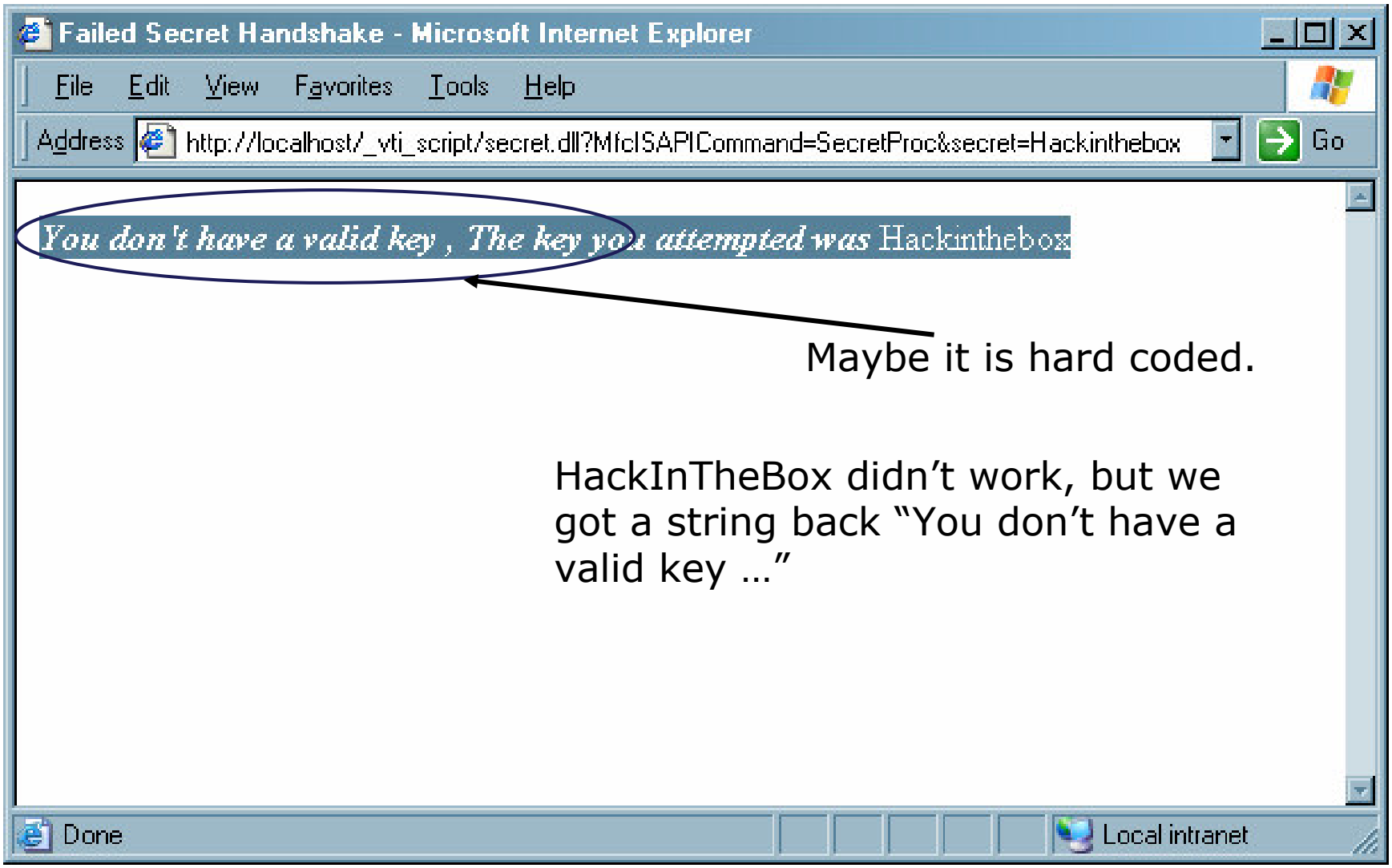
ISAPI – Web Interface



Lets attempt HackInTheBox as the key

Application Background: An ISAPI that expects a valid string to allow you to pass the "Gate" to get to the application.

ISAPI – Web Interface



Agenda

Product Testing Methodology

Basics

Debugging Steps

Ollydbg

Process Isolation

isDebuggerPresent

ISAPI Analysis

ISAPI – Debugging Steps

- Attach/Load ISAPI in debugger
  - Decide which debugger.
- Bypass any anti-debugging steps
- Step through binary
- Set breakpoints at best locations possible

ISAPI – Debugger, Ollydbg

- Lets attach a debugger to the DLL to see what is going on.
  - Ollydbg
  - GDB / DDD [GUID GDB]
  - Windbg
- Ollydbg is one of the best User Mode debuggers available and it is free
- To learn more about ollydbg, download and read the help files. Additional links will be provided at the end of the presentation.
- Ollydbg can be downloaded from http://www.ollydbg.de/
- Ollydbg has a community forum which has moved to this http://www.asmcommunity.net/board/. The older messages will be available sometime in the near future

Agenda

Product Testing Methodology

Basics

Debugging Steps

Ollydbg

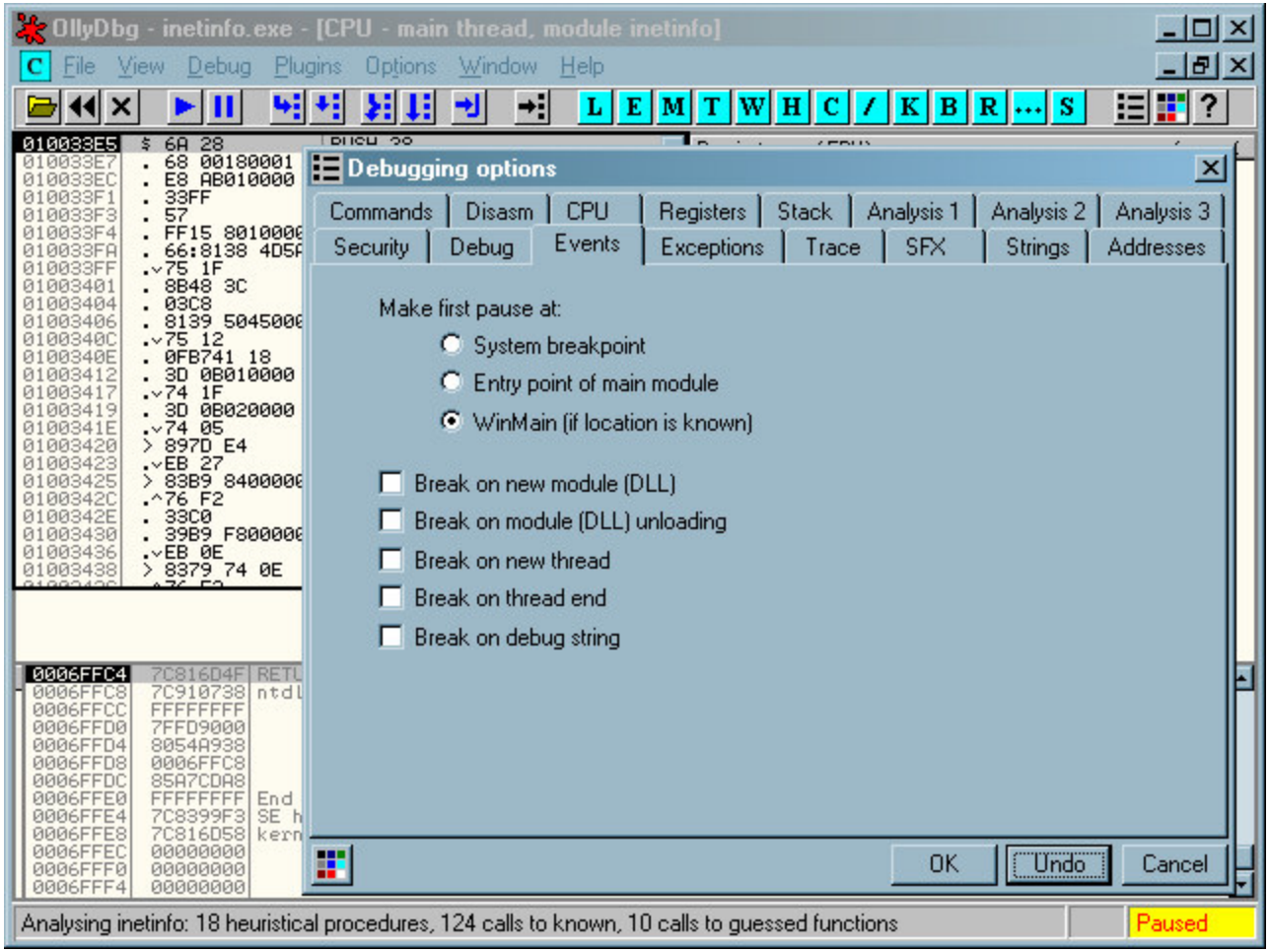
Process Isolation

isDebuggerPresent

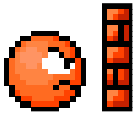
ISAPI Analysis

ISAPI - OllyDBG

- Attach inetinfo.exe process. [Demo]
- The default in OllyDbg breaks at WinMain. We shall wait for it to pause at that location.



ISAPI - Ollydbg



- View Executable Modules.
- The ISAPI can't be found ☹
- What should be our next step ?
- Ideas / Suggestions ?
- Search Google / Microsoft what is happening behind the scenes ?

Executable modules

Base	Size	Entry	Name	File version	Path
5A800000	00010000	5A80B4F1	wamreg	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\wamreg.dll
5A850000	00015000	5A859D50	wam	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\wam.dll
5AA20000	0005F000	5AA5D6AB	w3svc	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\w3svc.dll
5AD70000	00038000	5AD71626	uxtheme	6.00.2900.2180	C:\WINDOWS\system32\uxtheme.dll
5B860000	00054000	5B8689F8	NETAPI32	5.1.2600.2180 ((	C:\WINDOWS\system32\NETAPI32.dll
5BC20000	0000F000	5BC28505	svcext	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\svcext.dll
5BD00000	00006000	5BD01149	STAXMEM	6.0.2600.2180 ((	C:\WINDOWS\system32\STAXMEM.dll
5BE70000	0000F000	5BE7363C	sspifilt	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\sspifilt.dll
5D090000	00097000	5D0932DA	comctl_1	5.82 (xpsp_sp2_	C:\WINDOWS\system32\comctl32.dll
5D360000	0000E000		MFC71ENU	7.10.3077.0	C:\WINDOWS\system32\MFC71ENU.DLL
5D900000	00005000	5D901075	rpcref	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\rpcref.dll
5E060000	00006000	5E061B9B	pwsdata	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\pwsdata.dll
5F2F0000	0000D000	5F2F67C8	nsepm	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\nsepm.dll
61C20000	00017000	61C2F45B	metadata	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\metadata.dll
61DE0000	0000D000	61D54C5E	md5filt	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\md5filt.dll
629E0000	00007000	629E27EA	lonsint	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\lonsint.dll
65F00000	0000A000	65F043C4	iscomlog	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\iscomlog.dll
65F10000	00014000	65F1D2F4	ISATQ	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\ISATQ.dll
662B0000	00058000	662E7A51	hnetcfg	5.1.2600.2180 ((	C:\WINDOWS\system32\hnetcfg.dll
669F0000	00013000	669F998C	IISMAP	6.0.2600.2180 ((	C:\WINDOWS\system32\IISMAP.dll
66A10000	00017000	66A1B9AF	iislog	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\iislog.dll
66A50000	00005000	66A5175A	IISFECNV	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\IISFECNV.dll
66D00000	00004000	66D0479E	iisadmin	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\iisadmin.dll
67E70000	00005000	67E713D6	fpenedll	4.0.2.7523	C:\Program Files\Common Files\Microsoft Shared\Web Server Ext
689E0000	0000C000	689E7C20	gzip	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\gzip.dll
68A20000	00023000	68A330AC	IisRTL	6.0.2600.2180 ((	C:\WINDOWS\system32\IisRTL.DLL
694B0000	0000C000	694B13E5	exstrace	5.6.2600.2180 ((	C:\WINDOWS\system32\exstrace.dll
6E6C0000	00009000	6E6C4839	compfilt	5.1.2600.2180 ((	C:\WINDOWS\system32\inetsrv\compfilt.dll
6E780000	0000E000	6E788A35	COADMIN	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\COADMIN.dll
6F050000	00047000	6F059262	httpext	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\httpext.dll
71440000	0000E000	71445E14	ADMWPROX	6.0.2600.2180 ((	C:\WINDOWS\system32\ADMWPROX.dll
71470000	00008000	71476703	admexs	6.0.2600.2180 ((	C:\WINDOWS\system32\inetsrv\admexs.dll
71A50000	0003F000	71A514CD	mswsock	5.1.2600.2180 ((	C:\WINDOWS\system32\mswsock.dll
71A90000	00008000	71A9142E	wshtcpip	5.1.2600.2180 ((	C:\WINDOWS\System32\wshtcpip.dll
71AA0000	00008000	71AA1642	WS2HELP	5.1.2600.2180 ((	C:\WINDOWS\system32\WS2HELP.dll
71AB0000	00017000	71AB1273	WS2_32	5.1.2600.2180 ((	C:\WINDOWS\system32\WS2_32.dll
71AD0000	00009000	71AD1039	WSOCK32	5.1.2600.2180 ((	C:\WINDOWS\system32\WSOCK32.dll
71BF0000	00013000	71BF118D	SAMLIB	5.1.2600.2180 ((	C:\WINDOWS\system32\SAMLIB.dll
71F80000	00004000	71F81057	Security	5.1.2600.2180 ((	C:\WINDOWS\system32\Security.dll
73DD0000	000FE000	73DD73C8	MFC42	6.02.4131.0	C:\WINDOWS\system32\MFC42.DLL
74320000	0003D000	7432F659	ODBC32	3.525.1117.0 (w	C:\WINDOWS\system32\ODBC32.DLL
74380000	0000F000	74388AD9C	wdigest	5.1.2600.2180 ((	C:\WINDOWS\system32\wdigest.dll
745E0000	002C6000	745E3065	msi	3.1.4000.2435	C:\WINDOWS\system32\msi.dll
750B0000	00012000	750B12A4	RESUTILS	5.1.2600.2180 ((	C:\WINDOWS\system32\RESUTILS.DLL
750F0000	00013000	750F12A5	MTXCLU	2001.12.4414.30	C:\WINDOWS\system32\MTXCLU.DLL
75130000	00014000	751314A6	colbact	2001.12.4414.30	C:\WINDOWS\system32\colbact.DLL
753E0000	0006D000	75409342	VSSAPI	5.1.2600.2180 ((	C:\WINDOWS\system32\VSSAPI.DLL
763B0000	00049000	763B1AB8	comdlg32	6.00.2900.2180	C:\WINDOWS\system32\comdlg32.dll
76620000	0013C000	766240ED	comsvcs	2001.12.4414.30	C:\WINDOWS\system32\comsvcs.dll
767A0000	00013000	767A1250	NTDSAPI	5.1.2600.2180 ((	C:\WINDOWS\system32\NTDSAPI.dll
767F0000	00027000	767F146D	schannel	5.1.2600.2180 ((	C:\WINDOWS\system32\schannel.dll
769C0000	000B3000	769C15D4	USERENV	5.1.2600.2180 ((	C:\WINDOWS\system32\USERENV.dll
76B20000	00011000	76B2A0E0	ATL	3.05.2284	C:\WINDOWS\system32\ATL.DLL
76C30000	0002E000	76C31529	wintrust	5.131.2600.2180	C:\WINDOWS\system32\wintrust.dll
76C90000	00028000	76C9126D	IMAGEHLP	5.1.2600.2180 ((	C:\WINDOWS\system32\IMAGEHLP.dll
76D10000	00011000	76D111D9	CLUSAPI	5.1.2600.2180 ((	C:\WINDOWS\system32\CLUSAPI.DLL
76D60000	00019000	76D653F7	iphlpapi	5.1.2600.2180 ((	C:\WINDOWS\system32\iphlpapi.dll
76F20000	00027000	76F28368	DNSAPI	5.1.2600.2180 ((	C:\WINDOWS\system32\DNSAPI.dll
76F60000	0002C000	76F61130	WLDAP32	5.1.2600.2180 ((	C:\WINDOWS\system32\WLDAP32.dll
76FD0000	0007F000	76FD3115	CLBCATQ	2001.12.4414.30	C:\WINDOWS\system32\CLBCATQ.DLL
77050000	000C5000	77051055	COMRes	2001.12.4414.25	C:\WINDOWS\system32\COMRes.dll
77120000	0008C000	77121558	OLEAUT32	5.1.2600.2180	C:\WINDOWS\system32\OLEAUT32.dll
773D0000	00102000	773D42B3	comctl32	6.0 (xpsp_sp2_r	C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-Controls_6595b
774E0000	0013D000	774FD0A1	ole32	5.1.2600.2726 ((	C:\WINDOWS\system32\ole32.dll
77A80000	00094000	77A81642	CRYPT32	5.131.2600.2180	C:\WINDOWS\system32\CRYPT32.dll
77B20000	00012000	77B23399	MSASN1	5.1.2600.2180 ((	C:\WINDOWS\system32\MSASN1.dll
77C00000	00008000	77C01135	VERSION	5.1.2600.2180 ((	C:\WINDOWS\system32\VERSION.dll
77C10000	00058000	77C1F2A1	msvcrt	7.0.2600.2180 ((	C:\WINDOWS\system32\msvcrt.dll
77D40000	00090000	77D4F538	USER32	5.1.2600.2622 ((	C:\WINDOWS\system32\USER32.dll
77DD0000	0009B000	77DD70D4	ADVAPI32	5.1.2600.2180 ((	C:\WINDOWS\system32\ADVAPI32.dll
77E70000	00091000	77E76284	RPCRT4	5.1.2600.2180 ((	C:\WINDOWS\system32\RPCRT4.dll
77F10000	00047000	77F165BA	GDI32	5.1.2600.2818 ((	C:\WINDOWS\system32\GDI32.dll
77F60000	00076000	77F651FB	SHLWAPI	6.00.2900.2861	C:\WINDOWS\system32\SHLWAPI.dll
77FE0000	00011000	77FE2131	Secur32	5.1.2600.2180 ((	C:\WINDOWS\system32\Secur32.dll
79E50000	00006000	79E51BCC	aspnet_f	1.1.4322.573	C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\aspnet_filter.dl
7C140000	00103000	7C14F41A	MFC71	7.10.3077.0	C:\WINDOWS\system32\MFC71.DLL
7C340000	00056000	7C34229F	MSVCR71	7.10.3052.4	C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\MSVCR71.dll
7C800000	000F4000	7C80B436	kernel32	5.1.2600.2180 ((	C:\WINDOWS\system32\kernel32.dll
7C900000	000B0000	7C913156	ntdll	5.1.2600.2180 ((	C:\WINDOWS\system32\ntdll.dll
7C9C0000	00815000	7C9E7326	SHELL32	6.00.2900.2869	C:\WINDOWS\system32\SHELL32.dll

Agenda

Product Testing Methodology

Basics

Debugging Steps

Ollydbg

Process Isolation

isDebuggerPresent

ISAPI Analysis

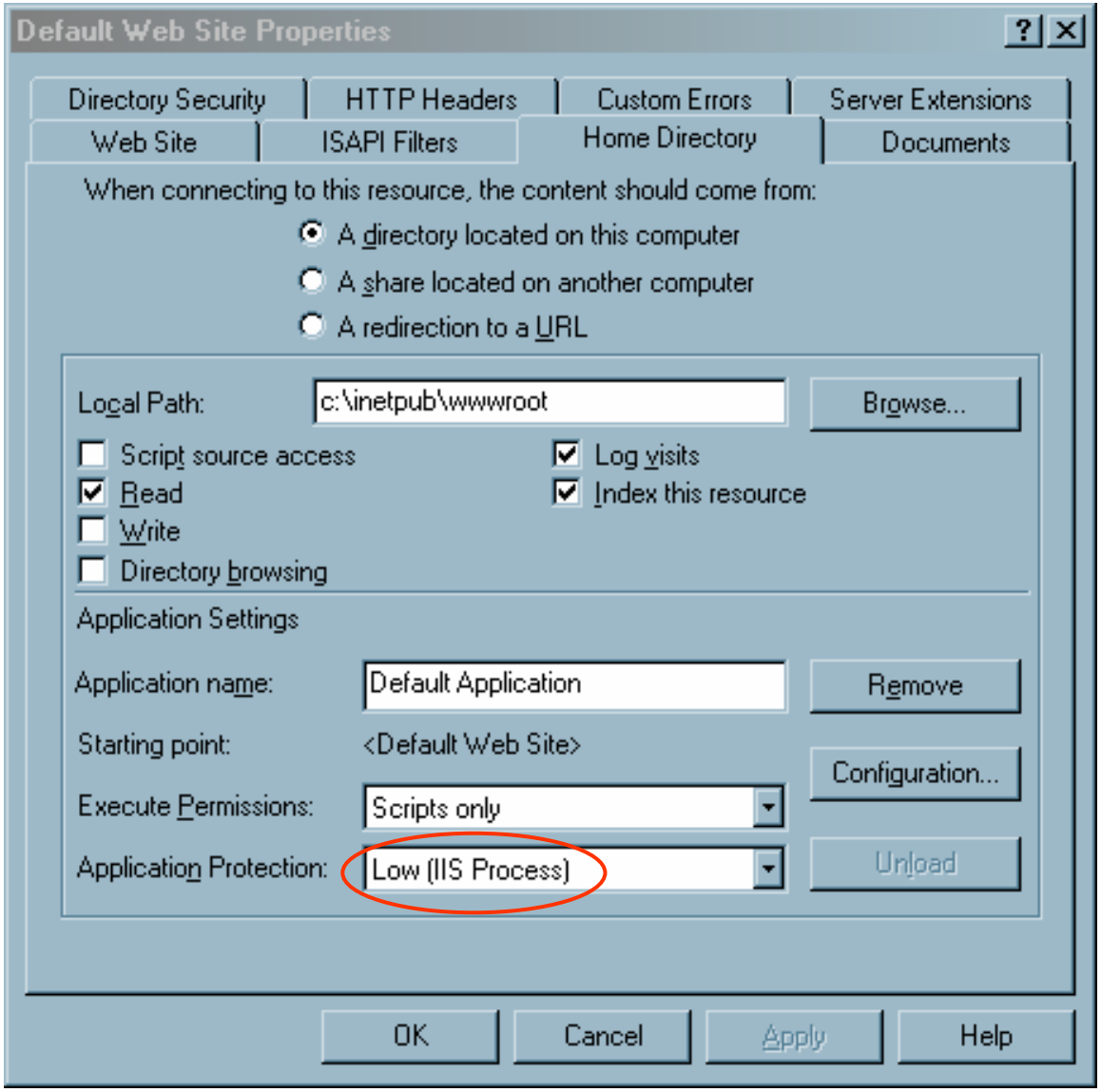
IIS Process Isolation - Microsoft

- Application Protection or Isolating Applications refers to the process in which applications (ISAPIs) are run. It means configuring them to run in a process (memory space) that is separate from the Web server and other applications. You can configure applications to have one of three levels of application protection:
  - Low (in-process) application protection.
  - Medium (pooled) application protection.
  - High (isolated) application protection.

- IIS 5.0 & 6.0 offers three levels of application protection.
  - Low - ISAPI is inside inetinfo.exe process space
  - Medium – ISAPI is in dllhost.exe process space
  - High – ISAPI is in dllhost.exe process space.

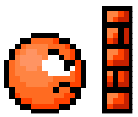
- In case you encounter a In IIS 4.0 based application, the ISAPI ran in either (Inetinfo.exe) or in a process separate from Web services (DLLHost.exe), Low & High only.

IIS Process Isolation - Microsoft



ISAPI – Ollydbg

- Easiest for debugging is make it low; in-process application protection. We might have other applications running in DLLhost.exe process space and we don't want to kill our other apps.
- Restart IIS
- Attach Inetinfo.exe and let the application loop through while you input data into the HTML form
- Should be able to find Secret.dll in loaded executables (View / Executable Modules) in inetinfo.exe



- Application is Terminated
- IDEAS / Suggestions ?

Agenda

Product Testing Methodology

Basics

Debugging Steps

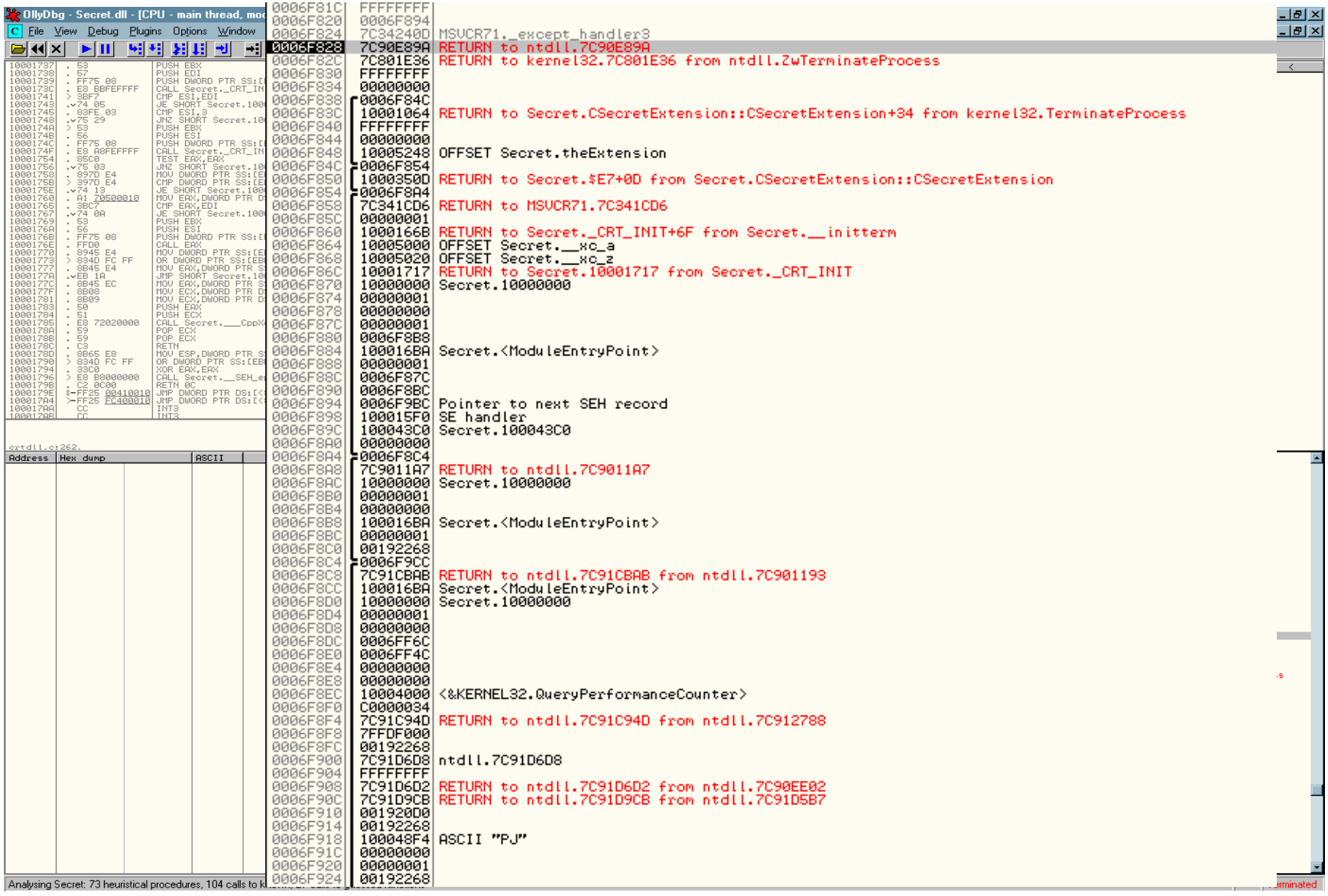
Ollydbg

Process Isolation

isDebuggerPresent

ISAPI Analysis

ISAPI – isDebuggerPresent()



ISAPI – isDebuggerPresent()

- ntdll.ZwTerminateProcess
- Viewing names in Secret.dll we find that there is a call to function isDebuggerPresent.
- The IsDebuggerPresent function terminates execution of the process. This is a common technique used to discourage debugging of applications.

Manually - Bypassing iSDebuggerPresent

Method 1

- Set a breakpoint on the IsDebuggerPresent function.
  - we'll load the command-line plug-in (Alt+F1)
  - *bp IsDebuggerPresent*
- Once the break point is reached, executes "Step Into" twice (Shift+F7 * 2) and stop
- By right clicking in the Disassembler pane and selecting Follow in Dump | Memory Address, the location and value of the IsDebuggerPresent function is displayed in the Dump pane. The location is 7FFDA002
- 01 00 FF FF FF FF 00 00 40 00 A0 1E 19 00
- Right-clicking the first value in this string (01) and selecting "Binary\Fill with 00's"

Manually - Bypassing iSDebuggerPresent

Method 2

- A simpler way to do this is to load Ollydbg's command-line plug-in (Alt-F1).
- Insert the following command

  set byte ptr ds:[fs:[30]+2]] = 0
- The command changes the return value of the API (isDebuggerPresent) to always be 0.

Plugins - Bypassing isDebuggerPresent

- Hide Debugger
  - Hides debugger from the Kernel32.isDebuggerPresent function
- isDebuggerPresent Plugin
  - Hide mode, hides debugger from the Kernel32.isDebuggerPresent function
  - Extra Hide
  - Dump process (need to provide from where to where).

Plugins - Bypassing isDebuggerPresent

- Olly Invisible
  - System Wide Hooking - This method Ollydbg will be hidden to the whole system.
  - User Wide Hooking - This method Ollydbg will be hidden to the current user session only.
  - Only Target Process – This method hides to targe process only (saves hooking)

- Functions Olly Invisible hides

  - IsdebuggerPresent
  - BeingDebugged
  - CsrGetProcessId
  - ZwQuerySystemInformation
  - ZwQueryInformationProcess

Agenda

Product Testing Methodology

Basics

Debugging Steps

Ollydbg

Process Isolation

isDebuggerPresent

ISAPI Analysis

ISAPI - Analysis

- Enumerate functions, imported and exported by ISAPI
- Enumerate strings inside ISAPI
- Review code of key functions
- Set breakpoint on any or all key aspects

ISAPI - Analysis

Enumerate functions, imported and exported by ISAPI

- The functions can be enumerated using Quick View or Quick View Plus utility
- Same can be done using dumpbin utility, provided with Visual Studio

  dumpbin /EXPORTS secret.dll
- In Ollydbg, to view the list of functions, select View Executable Modules, right-clicking on Secret.dll option from the list and selecting view names displays the functions being imported and exported

ISAPI - Analysis

Enumerating strings

- All the strings being used inside secret.dll can be viewed either by using the “strings” utility available from sysinternals website
- ASCII strings inside secret.dll can be viewed by right-clicking inside the Disassembler pane where secret.dll is loaded and selecting Search for | All referenced Text strings
- Set a breakpoint on any interesting strings that are seen. Example: “You don’t have a valid key, The key you attempted was”. The error we had seen on attempt to login.

ISAPI - Analysis

Review code of key functions

- Key Functions such as strcpy and strcat are being used

- To select references on import, right click on the function and select references on import. A new pane with a list of references pops up. Set a breakpoint on the references

ISAPI - Analysis

- Browse to the website and provide a string.
  - String providing strategy
    - AAAAAAAAAAAAAAA
    - AAAABBBBCCCCDDDD
    - 1111222233334444

- The application should stop before the Failed secret error message.

- Tracing the call in code a few lines above the breakpoint, we note a comparison is performed between the data input and a string locally stored in the binary, “SecurityCompass”.

- GAME OVER !!

ISAPI - Input Valid String

- http://localhost/ and providing the string SecurityCompass.

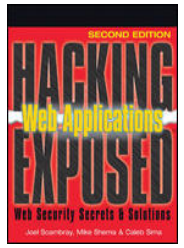
- To obtain the dll either browse to securitycompass.com and obtain it from the resources section or by browsing to http://www.webhackingexposed.com/.
  Available only after July 01, 2006.

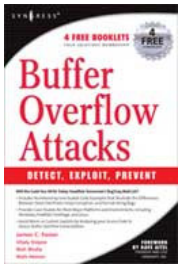
Some Interesting Links

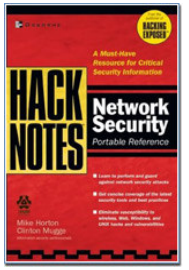
- OllyDebugger Web Site
  http://www.ollydbg.de/
- OllyDebugger Forum for Discussion
  http://www.asmcommunity.net/board/
- Open Reverse Code Engineering Community Website
  http://www.openrce.org/
- Site with a lot of interesting tools
  http://www.sysinternals.com/Utilities/Strings.html
- Assembly Tutorials
  http://spiff.tripnet.se/~iczelion/tutorials.html

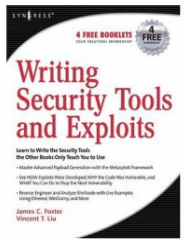
Security Compass Profile

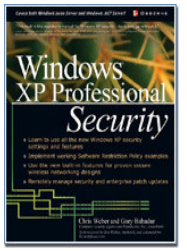
- Our consultants have serviced large (Fortune 500) and medium sized companies across most major industries
- We have worked for major security players, including Foundstone and Deloitte
- We have co-authored or contributed to several security books, including:
  - Buffer Overflow Attacks: Detect, Exploit & Prevent
  - Windows XP Professional Security
  - HackNotes: Network Security
  - Writing Security Tools and Exploits
  - Hacking Exposed: Web Applications, 2nd Edition
- We have presented at and continue to present at security conferences, including:
  - Reverse Engineering Conference 2005 in Montreal; HackInTheBox 2005 in Malaysia; ISC2's Infosec Conferences in Las Vegas, NYC, Toronto & DC; CSI NetSec; DallasCon; ToorCon; and Freenix.
- We present and contribute to open source projects:
  - Chair at OWASP Toronto, Presented at OWASP Toronto, Contributed to YASSP Project (Lead by SANS and Xerox), Botan Crypto library, Cutlas P2P network & VNCCrack











QUESTIONS ?

- Contact :
  Nish Bhalla (Nish@securitycompass.com)
  Founder, Security Compass
  Toronto, Ontario, CANADA: 647.722.4883
  Shrewsbury, New Jersey, USA: 201.390.9198