Remote Binary Planting
The Forgotten Vulnerability Affair
HITBSECCONF 2010 Malaysia

Mitja Kolsek
ACROS d.o.o.
mitja.kolsek@acrossecurity.com
www.acrossecurity.com
Objectives

1. Can we find 512+ bugs?
2. Can balloons be used as progress bar?
200 Bugs Milestone
The Life of Binary Planting


DLL Spoofing

“Double clicking on MS Office documents from Windows Explorer may execute arbitrary programs in some cases.”

2000  Georgi Guninski: Two Office bugs

2001  Nimda uses “DLL spoofing” for propagation

2004  Microsoft introduces “safe search order”

2005  “DLL Spoofing in Windows” paper (local attack)

2008  David LeBlanc: “DLL Preloading Attacks” paper

May 2009  Acros reports BP bugs to VMware

Mar 2010  Acros reports BP bugs to Apple, Google, Microsoft

Apr 2010  Phone conference with Microsoft

Meanwhile...  Microsoft preparing countermeasures

520+ bugs in stock

Aug 18, 2010  Apple fixes iTunes, Acros publishes ASPR

Later that day  The cat gets “out of the bug”
DLL Search Order – The “Troublemaker”

LoadLibrary(“SomeLib.dll”)

1. The directory from which the application loaded
2. Current Working Directory (CWD)
3. C:\Windows\System32
4. C:\Windows\System
5. C:\Windows
6. System PATH; User PATH
Causes For Not Finding Binaries in Primary Locations

1. Programmer checks for local capabilities by trying to load a library
2. Language-dependent DLLs
3. A custom/partial install
4. Application is prepared for future enhancements
5. Backward compatibility
6. O/S Porting (loading “linuxlib.so.1” on Windows)
7. Missing delay-load DLLs
8. Wrong assumptions about “side by side” DLLs
9. Some DLLs are present on OS1 but not on OS2 (dwmapi.dll)
10. Application written so that it finds its binaries in PATH
11. Different paths to system DLLs in registry between OS1 and OS2
12. Assumptions about installed components
13. DLL loaded by 3rd party process in another location
14. Incomplete uninstalls
15. ...

Closed-Source 3rd Party Components
Binary Planting Attacks
3-Step Attack Scenario

1. Planting a malicious binary
2. Getting CWD to the location of binary
3. Waiting for the app to load and execute it
Setting The Current Working Directory

1. Double-clicking a file in Explorer
2. File Open, File Save dialogs
3. Last open/save location
4. Fixed location
5. cmd.exe: cd command
6. File explorers
7. CreateProcess, ShellExecute
8. New process gets parent’s CWD
Internal Network Attack

1. Attacker
2. User
3. File Server

Internal Corporate Network
Local Goes Remote
Attacking From Internet – The WebDAV Magic

1. Attacker
2. Internet
3. Internal Corporate Network
4. User

»Hi John! Check out this document.«
Attack Vectors

1. Clicking on a link in browser
2. Clicking on a link in e-mail
3. Clicking on a link in IM message
4. Planting a binary on a file server
5. Document and binary in a ZIP archive
6. Document and binary on a USB stick
7. Document and binary on CD/DVD
8. Local privilege escalation
9. Advanced binary planting attacks
DLL Planting Demo

wab.exe
Address Book
Microsoft Corporation
Binary Planting Goes “EXE”
Searching for Non-Absolute EXEs

CreateProcess(“SomeApp.exe”)

1. The directory from which the application loaded
2. Current Working Directory (CWD)
3. C:\Windows\System32
4. C:\Windows\System
5. C:\Windows
6. System PATH; User PATH
Searching for Non-Absolute EXEs

`ShellExecute("SomeApp.exe")`

The directory from which the application loaded:

1. Current Working Directory (CWD)
2. C:\Windows\System32
3. C:\Windows\System
4. C:\Windows
5. System PATH; User PATH
Searching for Non-Absolute EXEs

_spawn*p* and _exec*p*

The directory from which the application loaded:

1. Current Working Directory (CWD)
2. C:\Windows\System32
3. C:\Windows\System
4. C:\Windows
5. System PATH; User PATH
Score

DLL Planting: 400+
EXE Planting: 120+
Our Research
Research Summary

Inspected 200+ Windows applications
   At least one exploitable Binary Planting issue in almost every one!
   (And we barely scratched the surface)

Recorded 520+ Binary Planting issues

Tool for detecting Binary Planting vulnerabilities
   GUI, monitoring processes
   Automated exploitation
   Ability to directly debug vulnerable code
### ACROS Binary Planting Detector

<table>
<thead>
<tr>
<th>Time Stamp</th>
<th>Process</th>
<th>Event Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>13:01:12:09</td>
<td>C:\Program Files\Google\Google Earth\GoogleEarth.exe</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Windows\Explorer.EXE</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Windows\Explorer.EXE</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Set CW0</td>
</tr>
<tr>
<td>13:01:20:29</td>
<td>C:\Documents and Settings\Local Settings\Application Data</td>
<td>Process detached</td>
</tr>
</tbody>
</table>

The page contains a screenshot of the ACROS Vulnerability Detector 0.9.1 interface, showing a table of events and process details. The interface includes options for monitoring, global attach, and event types, with a list of events such as process creation, detach, and execution details.
How Many Bugs?!?

XP ~1340m, Vista ~400m, Windows 7 ~150m
~100,000,000,000 bugs
Approx. 11,000 times the number of bicycles in Beijing
Hundreds of BP bugs on every Windows computer
Tens of thousands of ways to break into any bank
... or competitor’s network
... or government agency
... or nuclear facility
Affected Vendors

Microsoft
Apple
Google
VMware
IBM
Siemens
Mozilla
Adobe
Avast
Autodesk
Sophos
PGP ...

... 70+ at Secunia

...100+ from our research
Recommendations
Recommendations for Developers

• Use absolute paths to libraries and executables
• Don’t make “let’s see if it’s there” LoadLibrary* calls
• Don’t plan on finding your DLL/EXE in CWD or PATH
• Set CWD to a safe location at startup
• Use SetDllDirectory(“”) at startup
• Don’t use SearchPath function for locating DLLs
• Check your product with Process Monitor or another tool
• Test with CWDIllegalInDllSearch hotfix set to "max".
• Do this for all modules of your product!

http://www.binaryplanting.com/guidelinesDevelopers.htm
Recommendations for Administrators

- Install Microsoft’s Hotfix, remember to configure it
- Disable “Web Client” service
- Windows Software Restriction Policy, Windows AppLocker (DLL)
- Use a personal firewall with process and connection blocking
- Block outbound SMB on corporate firewall
- Block outbound WebDAV on corporate firewall
- Limit internal SMB, WebDAV traffic
- Restrict write access on file repositories to prevent planting
Recommendations for Users

• Be careful when using USB sticks, CDs, DVDs from unknown sources

• Think before double-clicking on anything presented to you

• If in doubt, download the data file (alone) to local drive and open it

• Alert your administrators about binary planting
What Microsoft Could Do

**Short Term**

- Extend the hotfix to EXE
- Introduce SetExeDirectory()
- Safe search path for EXE loading
- Set the default for file browse dialogs to not change CWD

**Long Term**

- Remove CWD from search paths
The Ultimate Solution: Eliminating CWD From The Game
Apple Re-Hacking Demo
DLL Search Order after SetDllDirectory Call

```
SetDllDirectory(safepath)
LoadLibrary("SomeLib.dll")
```

1. The directory from which the application loaded
2. C:\Windows\System32
3. C:\Windows\System
4. C:\Windows
5. The `SetDllDirectory` location
6. System PATH; User PATH
Unresolved Environment Variables

• “Win32 Oddities – Unable to Expand System Variables”
• “Vista - REG_EXPAND_SZ only seems to expand some variables”
• “Path Environment Variable Incorrect After Logon”
• “Windows installer screws up the PATH environment variable”
• “Environment variables not being expanded in Path registry entry”
• “ExpandEnvironmentStringsForUser() API does NOT expand the environment variable %USERNAME% on Windows 7”

• Microsoft Support, 2007: “Environment variable may not expand %APPDATA% to the Application folder”

http://support.microsoft.com/kb/329308
Unresolved Environment Variables – Real World Examples

- %APPDATA%/Python/Scripts
- %ProgramFiles(x86)%
- %CommonProgramFiles%/Microsoft Shared/Windows Live
- %PROGRAMFILES(x86)%/Common Files/Microsoft Shared/Ink
- %USERPROFILE%/Local Settings/Temp
- %systemroot%/system32/DATA/Config
- %NpmLib%
0-Code
Vulnerable Application Demo
1. Minimizing social engineering ("drive by")
2. Attacks on servers
3. New attack vectors
4. Bypassing BP protection
5. Wormification (beyond Stuxnet)

Select your favorite planting method

- DLL Planting
- EXE Planting
Resources

www.binaryplanting.com
blog.acrossecurity.com

http://support.microsoft.com/kb/2264107
http://blog.metasploit.com/2010/08/better-faster-stronger.html
http://securityxploded.com/dllhijackauditor.php

http://secunia.com/advisories/windows_insecure_library_loading/

Google “binary planting”, “dll hijacking”, “dll preloading”
Public Binary Planting Tools

DLLHijackAuditKit
Are you Binary Planting positive?

www.binaryplanting.com/test.htm

(tell your friends, colleagues about it)