Defending Microsoft Windows against 0-day exploits using EMET

ICSJWG – Fall 2012

Michael Orlando
[mforlando@cert.org]
NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.
Outline

• Overview of me - Vulnerability Analysis
  
• Bugs are everything and easy to find
  — Fuzz testing tools are publicly available (BFF/FOE)

• I am fully patched so I am safe right?
  — 0-days

• Possible exploitation protections
  — DEP
  — ASLR

• Microsoft EMET
  — ROP Mitigations

• Demo
Introduction

Michael Orlando
CERT/CC Vulnerability Analysis Team

• Analysis and research
• Coordination and disclosure
  — Vendors, researchers, other CSIRTs (including ICS-CERT)
• Discovery
  — Tools and methods to find vulnerabilities
Vulnerabilities/Exploits
Fuzzing

Everything is vulnerable

- Dumb fuzzing has found vulnerabilities in everything we’ve targeted
- We (and others) have been focusing on common, complicated binary formats
  - PDF
  - Office document formats
  - Flash
CERT Fuzzing Tools

Dranzer: Smart ActiveX fuzzer

File format fuzzers

- BFF: Basic Fuzzing Framework (Linux/MacOSX)
- FOE: Failure Observation Engine (Windows)
- Most effective against uncompressed binary formats
## CVEs Assigned

<table>
<thead>
<tr>
<th>Year</th>
<th>NVD CVE Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1998</td>
<td>0</td>
</tr>
<tr>
<td>1999</td>
<td>1573</td>
</tr>
<tr>
<td>2000</td>
<td>1236</td>
</tr>
<tr>
<td>2001</td>
<td>1538</td>
</tr>
<tr>
<td>2002</td>
<td>2368</td>
</tr>
<tr>
<td>2003</td>
<td>1495</td>
</tr>
<tr>
<td>2004</td>
<td>2629</td>
</tr>
<tr>
<td>2005</td>
<td>4601</td>
</tr>
<tr>
<td>2006</td>
<td>6975</td>
</tr>
<tr>
<td>2007</td>
<td>6429</td>
</tr>
<tr>
<td>2008</td>
<td>6981</td>
</tr>
<tr>
<td>2009</td>
<td>4817</td>
</tr>
<tr>
<td>2010</td>
<td>4663</td>
</tr>
<tr>
<td>2011</td>
<td>3661</td>
</tr>
<tr>
<td>2012</td>
<td>620*</td>
</tr>
</tbody>
</table>
## Microsoft Patch Tuesday

<table>
<thead>
<tr>
<th>Date</th>
<th>Bulletin Number</th>
<th>KB Number</th>
<th>Title</th>
<th>Bulletin Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/14/2012</td>
<td>MS12-060</td>
<td>2720573</td>
<td>Vulnerability in Windows Common Controls Could Allow Remote Code Execution</td>
<td>Critical</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-059</td>
<td>2733918</td>
<td>Vulnerability in Microsoft Visio Could Allow Remote Code Execution</td>
<td>Important</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-058</td>
<td>2740358</td>
<td>Vulnerability in Microsoft Exchange Server WebReady Document Viewing Could Allow Remote Code Execution</td>
<td>Critical</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-057</td>
<td>2731879</td>
<td>Vulnerabilities in Microsoft Office Could Allow for Remote Code Execution</td>
<td>Important</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-056</td>
<td>2706045</td>
<td>Vulnerability in JScript and VBScript Engines Could Allow Remote Code Execution</td>
<td>Important</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-055</td>
<td>2731847</td>
<td>Vulnerabilities in Windows Kernel-Mode Drivers Could Allow Elevation of Privilege</td>
<td>Important</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-054</td>
<td>2733594</td>
<td>Vulnerabilities in Windows Networking Components Could Allow Remote Code Execution</td>
<td>Critical</td>
</tr>
<tr>
<td>8/14/2012</td>
<td>MS12-053</td>
<td>2723135</td>
<td>Vulnerability in Remote Desktop Could Allow Remote Code Execution</td>
<td>Critical</td>
</tr>
</tbody>
</table>
Security updates available for Adobe Reader and Acrobat

• **Release date:** April 10, 2012
• **Last updated:** April 17, 2012
• **Vulnerability identifier:** APSB12-08
• **Vulnerability Summary for CVE-2012-0775**
  • **Impact**
    • CVSS Severity (version 2.0):
    • CVSS v2 Base Score: 10.0 (HIGH) (AV:N/AC:L/Au:N/C:C/I:C/A:C) (legend)
    • Impact Subscore: 10.0
    • Exploitability Subscore: 10.0
I am patched so I must be safe

I have applied all of vendor xyz’s patches so I am safe right?
Internet Explorer 0-day anyone?

• Microsoft Internet Explorer 6/7/8/9 contain a use-after-free vulnerability, CVE-2012-4969
  • http://www.kb.cert.org/vuls/id/480095
0-Day isn’t Rare
Exploitation Protections
Exploiting vulnerabilities

Get control of Instruction Pointer (EIP)

- Control of EIP == Control of execution
- Point EIP to attacker’s code (shellcode) : attacker’s code executes
Exploiting vulnerabilities

Memory layout:

- Application code
- Loaded Document
- Shellcode

EIP
Protection #1: DEP

Data Execution Prevention

- Do not execute memory locations that do not have execute permissions
- Requires processor support: NX bit
- Applications must opt-in
DEP Protection

Memory layout:

Application code (executable)

Loaded Document

Shellcode (not executable)

DEP: ON

DEP Violation
Program Terminated

EIP
Time to go home!

DEP solves the problem, right?
Return Oriented Programming

Use pieces of existing executable code to accomplish your goal of bypassing DEP. Several techniques can be used, including:

- Turn off DEP
- Mark memory as executable
- Allocate new executable memory
- Copy shellcode to executable memory

Outcome: Executable shellcode
Exploiting vulnerabilities

Memory layout:

Application code (executable)
- Turn Off DEP (executable)
- Loaded Document
- Shellcode (not executable)

EIP

DEP: ON DEP: OFF

Turn Off DEP
Protection #2: ASLR

Address Space Layout Randomization

- Executable modules loaded at randomized location
- Breaks ROP
Exploiting vulnerabilities

Memory layout:

- Turn Off DEP (executable)
- Application code (executable)
- Application code (executable)
- Turn Off DEP (executable)
- Loaded Document
- Shellcode
- Loaded Document (not executable)
- Shellcode (not executable)

Invalid Instruction
Program terminated

DEP: ON
ASLR: ON
Exploit Mitigation

DEP and full ASLR together help prevent exploitation of vulnerabilities.

- DEP without ASLR is not effective
  - Vista or later is required for ASLR
- ASLR without DEP is not effective
- Every loaded module needs to opt in to ASLR
Vulnerability Exploit protection

What do we know about vulnerability protection?

• Vendors don’t always opt in to exploit mitigations
• Vendors don’t fix known vulnerabilities in a timely manner
• We want protection from unknown vulnerabilities
Exploitation Protections - EMET
Microsoft EMET

Don’t be at the mercy of your software vendors. Microsoft Enhanced Mitigation Experience Toolkit can force-enable:

- DEP
- ASLR (Vista and newer)
- SEHOP
- Additional exploit mitigations

http://support.microsoft.com/kb/2458544
Microsoft EMET

- Can be force-enable:
  - System Wide
  - Application Based

- Can be deployed and configured over Group Policy and System Center Configuration Manager (EMET 3.0)
ASLR Requires Vista or Newer

Windows XP (Server 2003) does not support ASLR!
Microsoft EMET System Wide (XP)
Microsoft EMET System Wide (Vista+)*
Microsoft EMET Per Application

![Application Configuration](image)

<table>
<thead>
<tr>
<th>App Name</th>
<th>DEP</th>
<th>NullPage</th>
<th>HeapSpray</th>
<th>BottomUpASLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>IGSSdataServer.exe</td>
<td>![check]</td>
<td>![check]</td>
<td>![check]</td>
<td>![check]</td>
</tr>
<tr>
<td>IGSSMaster.exe</td>
<td>![check]</td>
<td>![check]</td>
<td>![check]</td>
<td>![check]</td>
</tr>
</tbody>
</table>
ROP Mitigations

EMET 3.5 introduces explicit ROP mitigations
Application Without EMET Mitigations
Application With EMET Mitigations
Use EMET to stay safe

The way to more safely run applications on Windows is to use EMET!

- Minimize risk of delayed patching
- Protect against known vulnerabilities
- Protect against 0day vulnerabilities
- Protect against future vulnerabilities
- EMET 3.5 ROP protection buys time for migration off of Windows XP
For More Information

Visit CERT® web sites:
http://www.cert.org/vuls/discovery/
http://www.cert.org/blogs/certcc/
https://www.cert.org/vuls/discovery/bff.html
https://www.cert.org/vuls/discovery/foe.html

Contact Presenter
Michael Orlando
mforlando@cert.org
(412) 268-4334

Contact CERT:
Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh PA 15213-3890