Secunia
Stay Secure

DEP/ASLR Implementation Progress in Popular Third-party Windows Applications

June 29th 2010

Alin Rad Pop, Senior Security Specialist
Secunia Research

Table of Contents

I. Introduction

Vulnerabilities that corrupt memory typically result in the execution of arbitrary code by redirecting the program flow to a writable memory area containing instructions defined by an attacker.

DEP (Data Execution Prevention) is a generic defensive measure that prevents the execution of writable memory, first added to Windows in XP Service Pack 2 in August 2004. In a default configuration, Windows applications have to inform the operating system that they want DEP enabled in their context. The most popular method of enabling DEP on pre-Vista systems is a call to the "SetProcessDEPPolicy()" function. On Windows Vista and newer systems, an executable can communicate its DEP choice by simply setting the NX_COMPAT (0x100) flag in the "DllCharacteristics" field of a PE (Portable Executable).

While DEP renders the exploit development process more complex and time consuming, code execution can still be achieved by constructing a chain of function calls to fixed executable addresses inside the address space of the target process (technique known as "return-into-libc"). In order to encumber such techniques, ASLR (Address Space Layout Randomization) was introduced with the release of Windows Vista in early 2007. If a PE file has the DYNAMIC_BASE (0x40) flag set in its "DllCharacteristics" field, the address at which the PE is loaded is randomized on each system boot. This significantly lowers the chances for a return-into-libc attack to succeed. However, note that a single module for which ASLR is not applied can at times be enough for code execution to succeed.
If both DEP and ASLR are correctly deployed, the ease of exploit development decreases significantly, code execution being sometimes impossible to achieve in one attempt.

This paper intends to depict the evolution and current state of DEP and ASLR support in popular third-party Windows applications.

Windows XP SP3 and Windows 7, both running on a 32-bit platform, were used as testing environments. Sixteen of the most popular* third-party Windows applications were tested. Only applications, which had at least one memory corruption vulnerability publicly disclosed within the last two years, were chosen.

While performing ASLR compatibility tests, if DEP was found to be disabled in a tested application, code execution was considered achievable regardless of ASLR support. ASLR is, therefore, irrelevant in such cases and was not tested in detail.

** Application popularity was determined by using Secunia PSI statistics for Windows 7 and Windows XP systems.*

II. DEP/ASLR Support History and Current State

1. Adobe Flash Player
The presence of DEP depends on the browser including the Flash Player plugin. Fairly recent exploitation techniques involving jumps into controlled data generated by the JIT (Just-In-Time) ActonScript compiler rendered the DEP state of the including process irrelevant. However, preliminary tests indicate that the latest 10.1.53.64 version defeats typical JIT spraying techniques.

All 9.x versions were found to have a fixed base address. The first 10.x version, released in October 2008, had a dynamic base defined, allowing ASLR to function properly.

2. Sun Java JRE
Java resources are loaded and processed in the java.exe process and not in the context of the browser opening a web page that embeds a Java applet.

DEP was found to be disabled in the java.exe executable included in the latest version of the software (6 Update 20). Furthermore, default libraries are loaded at fixed addresses.

3. Adobe Reader
The NX_COMPAT flag was found to be set in versions as early as 8.1.0 - released in 2007 (previous versions were not tested). This results in the activation of DEP when running on Windows 7 systems.

As previously noted, Windows XP does not interpret the NX_COMPAT flag, requiring additional actions to be performed. A call to "SetProcessDEPPolicy()" was only added in version 9.2.0 (March 2010), enabling DEP on Windows XP SP3 for the first time.

From an ASLR perspective, an increased number of dynamic libraries have adopted the dynamic base flag over time. The latest version (9.3.2) still leaves nppdf32.dll loaded at a fixed address contrary to the made progress. nppdf32.dll is only used in non-IE browsers when viewing PDFs online, affecting the ASLR state of the parent browser.

Another notable fact is that the inclusion of Adobe Flash Player in Adobe Reader 9.x can be used to defeat DEP, the previously noted JIT exploitation techniques being applicable.

4. Mozilla Firefox
NX_COMPAT was first set in version 3.0 (June 2008). Again, this is not sufficient for DEP compatibility on Windows versions before Vista.

A call to "SetProcessDEPPolicy()" was added to xul.dll in version 3.5.2 (July 2009) for the 3.5.x branch. Although the 3.0.13 version was released at the same time as 3.5.2, the 3.0.x branch correctly applied DEP on all systems only in the following 3.0.14 version, released approximately one month later.

ASLR cannot be properly applied to the latest 3.6.3 version, which has 10 DLLs with a fixed base address. The following DLLs have a pre-defined base address of 0x10000000:

- plds4.dll
- ssl3.dll

- plc4.dll
- nss3.dll
- smime3.dll
- nssutil3.dll
- softokn3.dll
- nspr4.dll
- nssckbi.dll
- freebl3.dll

As a consequence, nspr4.dll is always loaded at address 0x10000000.

5. QuickTime Player
DEP is not enabled in the latest 7.6.6 version when using QuickTime as a standalone application (i.e. when manually opening a movie file).

Although default libraries are randomized when loading a QuickTime movie inside a browser, AppleProResDecoder.qtx is loaded at fixed address 0x10000000 when playing a video file encoded with the Apple ProRes codec.

6. VLC Media Player
The latest 1.0.5 version does not have DEP enabled. Additionally, multiple libraries are loaded at fixed addresses when the VLC browser plugin is used, affecting the ASLR state of the parent browser.

7. Apple iTunes
DEP was first enabled on Windows 7 (Vista) via NXCOMPAT in version 7.7.0.43, released July 2008.

The latest 9.2.0.61 version has no DEP for Windows XP and ASLR is negatively affected by CoreFP.dll and libdispatch.dll, which have fixed addresses in a default instance.

8. Google Chrome
While DEP has been enabled on both Windows 7 (Vista) and Windows XP from the first 1.x stable releases (late 2008), the icudt42.dll library is loaded at fixed address 0x4AD00000 in version 4.1.249.1064. Other icudt*.dll versions are loaded at fixed addresses in previous versions. The first stable version to enable dynamic allocation of the library was 5.0.375.55, released May 2010.

9. Adobe Shockwave Player
DEP depends on the browser including the Shockwave Player plugin.

From an ASLR perspective, the latest version (11.5.7.609) includes static DIRAPI.dll (0x680000000) and IML32.dll (0x69000000) libraries as well as numerous default assets having a fixed base.

10. OpenOffice.org
DEP is not enabled in the latest version (3.2.1).

If the browser plugin is configured, so_activex.dll is loaded at fixed address 0x10000000 in Internet Explorer's space.

11. Google Picasa
DEP is not enabled in the latest version (3.6.0 Build 105.67).

12. Foxit Reader
DEP was found to be disabled in the latest 3.3.1.0518 version. npFoxitReaderPlugin.dll and FoxitReaderOCX.ocx have a fixed base of 0x10000000, both being loaded by Firefox when viewing a PDF online.

13. Opera
DEP was first enabled for Windows 7 (Vista) systems in version 9.64, released March 2009. For Windows XP systems, a call to "SetProcessDEPPolicy()" was added to opera.exe in version 10.01, released October 2009. Opera.dll is loaded at fixed address 0x674F0000 in the latest version (10.53), affecting ASLR.

14. Winamp
DEP is not enabled in the latest 5.572 version.

15. RealPlayer
DEP is not enabled in the latest SP 1.1.4 version.

If the "Web Download & Recording" feature is enabled, rpbrowserrecordplugin.dll is loaded at fixed address 0x60000000 in Internet Explorer. Similar plugins are loaded at fixed addresses in other installed browsers.

16. Apple Safari
DEP was first used in version 4.0 (June 2009), enabled for both Windows 7 (Vista) and XP systems .

Version 4.0.5 loads AppleVersions.dll at fixed address 0x10000000 by default. The most recent 5.0 version makes AppleVersions.dll dynamic, but loads libdispatch.dll at fixed address 0x10000000.

III. Conclusion

DEP and ASLR support, although usually trivial to implement, is overlooked by a large number of application developers. The requirement for an additional call to "SetProcessDEPPolicy()" proved confusing to almost all vendors, resulting in late implementation of DEP when running on Windows XP.

Some developers have over time made their applications compatible with DEP, but overall the implementation process has proven slow and uneven between OS versions. ASLR support is on the other hand improperly implemented by almost all vendors, allowing return-into-libc techniques to likely succeed in their applications or in browsers designed to be otherwise ASLR compliant.

While most Microsoft applications take full advantage of DEP and ASLR, third-party applications have yet to fully adapt to the requirements of the two mechanisms. If we also consider the increasing number of vulnerabilities discovered in third-party applications, an attacker's choice for targeting a popular third-party application rather than a Microsoft product becomes very understandable.

Hopefully, vendors will see the importance of properly deploying the two measures, resulting in an increased number of third-party applications having full DEP and ASLR support in the near future.

IV. Appendix – Deployment Snapshots

Application	DEP (7)	DEP (XP)	Full ASLR
Flash Player	N/A	N/A	no
Sun Java JRE	no	no	No
Adobe Reader	**YES**	no	no
Mozilla Firefox	no	no	no
Apple Quicktime	no	no	no
VLC Media Player	no	no	no
Apple iTunes	no	no	no
Google Chrome	N/A	N/A	N/A
Shockwave Player	N/A	N/A	no
OpenOffice.org	no	no	no
Google Picasa	no	no	no
Foxit Reader	no	no	no
Opera	no	no	no
Winamp	no	no	no
RealPlayer	no	no	no
Apple Safari	no	no	no

DEP & ASLR (May 2008)

Application	DEP (7)	DEP (XP)	Full ASLR
Flash Player	N/A	N/A	**YES***
Sun Java JRE	no	no	no
Adobe Reader	**YES***	no	no
Mozilla Firefox	**YES**	no	no
Apple Quicktime	no	no	no
VLC Media Player	no	no	no
Apple iTunes	**YES**	no	no
Google Chrome	**YES**	**YES**	no
Shockwave Player	N/A	N/A	no
OpenOffice.org	no	no	no
Google Picasa	no	no	no
Foxit Reader	no	no	no
Opera	**YES**	no	no
Winamp	no	no	no
RealPlayer	no	no	no
Apple Safari	no	no	no

DEP & ASLR (April 2009)

Application	DEP (7)	DEP (XP)	Full ASLR
Flash Player	N/A	N/A	**YES***
Sun Java JRE	no	no	no
Adobe Reader	**YES***	no	no
Mozilla Firefox	**YES**	**YES**	no
Apple Quicktime	no	no	no
VLC Media Player	no	no	no
Apple iTunes	YES	no	no
Google Chrome	**YES**	**YES**	no
Shockwave Player	N/A	N/A	no
OpenOffice.org	no	no	no
Google Picasa	no	no	no
Foxit Reader	no	no	no
Opera	**YES**	**YES**	no
Winamp	no	no	no
RealPlayer	no	no	no
Apple Safari	**YES**	**YES**	no

DEP & ASLR (October 2009)

Application	DEP (7)	DEP (XP)	Full ASLR
Flash Player	N/A	N/A	**YES**
Sun Java JRE	no	no	no
Adobe Reader	**YES***	**YES***	no
Mozilla Firefox	**YES**	**YES**	no
Apple Quicktime	no	no	no
VLC Media Player	no	no	no
Apple iTunes	YES	no	no
Google Chrome	**YES**	**YES**	**YES**
Shockwave Player	N/A	N/A	no
OpenOffice.org	no	no	no
Google Picasa	no	no	no
Foxit Reader	no	no	no
Opera	**YES**	**YES**	no
Winamp	no	no	no
RealPlayer	no	no	no
Apple Safari	**YES**	**YES**	no

DEP & ASLR (June 2010)

** Exploitation techniques defeating the feature are publicly known*

IV. References

Protecting Your Code with Visual C++ Defenses:
http://msdn.microsoft.com/en-us/magazine/cc337897.aspx#S3

Understanding DEP as a mitigation technology part 1:
http://blogs.technet.com/b/srd/archive/2009/06/05/understanding-dep-as-a-mitigation-technology-part-1.aspx

Getting around non-executable stack (and fix):
http://archives.neohapsis.com/archives/bugtraq/1997_3/0281.html