惡意程式 PlugX 的新型變種分析報告

原文出處：Naked Security

http://nakedsecurity.sophos.com/2013/12/04/new-plugx-malware-variant-takes-aim-at-japan/?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed:+nakedsecurity+(Naked+Security+-+Sophos)

我們之前已經在 Naked Security 上討論過了 PlugX 這個惡意程式，感謝來自 SophosLabs 首席研究員 Gabor Szappanos（Szappi）的兩篇報告。Szappi 在五月時介紹 PlugX 的 6 版，深入探討了這個有趣的惡意程式家族，並在七月介紹他的近親 Smoaler。

這些惡意程式樣本中使用的有趣的戰術包括：

- 他們散布在似乎可信的電子郵件內。
- 電子郵件內含有惡意的附件。
- 這附件會利用微軟 Word 在處理 RTF 檔案的漏洞。
- 該附件含有西藏(圖博)的政治議題。
- 惡意程式在數個階段中解壓縮自己。
- 惡意程式載入和執行有自己內容的執行檔，並繞過作業系統的程式載入器。
- 惡意程式攜帶了數位簽章合法的應用程式，用於啟動感染。
- 這個合法的應用程式被欺騙以載入惡意的 DLL。
- 最終惡意程式會打開一個後門，允許攻擊者遠端操控惡意程式。

PlugX 再臨

現在，Szappi 和他的同事，在雪梨的 SophosLabs 的 XinRan Wu，一直在研究一些似乎像是 PlugX 家族的其他惡意程式樣本。

他們所遇到的是相同點和不同點的一個奇怪的混合。

這種特殊的威脅還不是很廣泛：我們只有一份野生的報告，連同發出它的電子郵件，加上少數其他來源的樣本。

我們因此還不知道 PlugX 惡意軟體的作者是正在練習新的目標，或是實驗新的惡意軟體，還是（希望如此！）發現它這些日子以來越來越難以得到感染的立足點。

這次攻擊的主要區別是顯而易見的：

- 惡意程式針對日本。

- 西藏(圖博)主題被替換為管理主題。
- 惡意程式利用的漏洞來自日本流行的文字處理器—一太郎，而不是 Word。

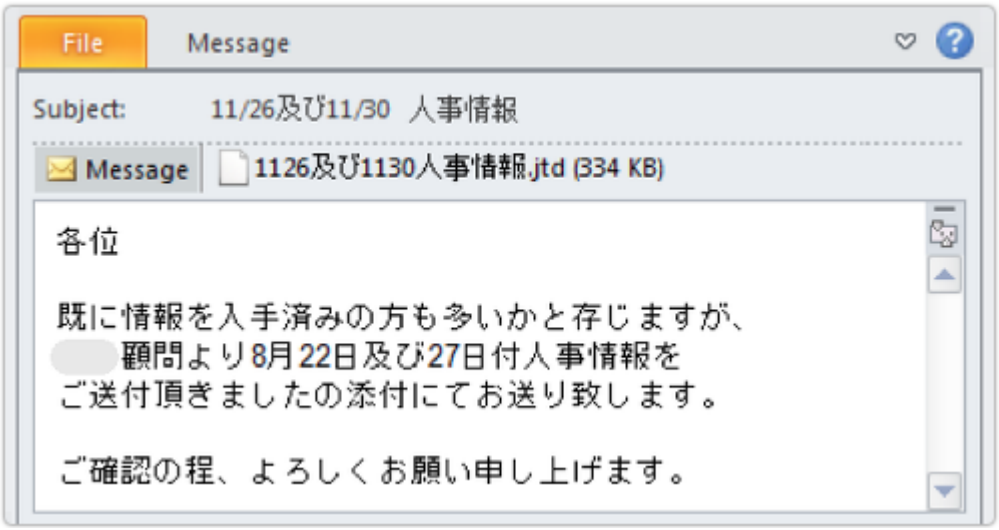
但有很多相似的地方：

- 惡意程式載入和執行有自己內容的執行檔，並繞過作業系統的程式載入器。
- 惡意程式攜帶了數位簽章合法的應用程式，用於啟動感染。
- 這個合法的應用程式被欺騙以載入惡意的 DLL。
- 最終惡意程式會打開一個後門，允許攻擊者遠端操控惡意程式。

惡意程式如何抵達

惡意程式被包在一封含有惡意附件的電子郵件內。

電子郵件看起來像這樣：



主題：11/26 與 11/30 的人事訊息

附件：11/26 與 11/30 的人事訊息.jtd(334KB)

內文：

各位

舊的訊息應該都已經有保存了，

但是我收到了來自 XX 顧問的 8 月 22 日及 8 月 27 日的人事訊息，並且轉寄給你們。

請您確認，非常感謝。

副檔名 JTD 由一太郎文件檔案所使用。

我們不能為您提供精確的細節關於該漏洞如何利用或如何降低它(我們沒有能夠使它觸發的一太郎環境)，但我們建議所有的一太郎用戶看看來自產品的供應商，Justsystems 公司的新安全公告。

公告詳細說明了最近的修補程序，以從受惡意附件危害的檔案中保護你，該漏洞已被記錄為 CVE-2013-5990：


セキュリティ情報

[JS13003]一太郎の脆弱性を悪用した不正なプログラムの実行危険性について

公開日：2013年11月12日

概要

弊社の一部製品に脆弱性の存在を確認いたしました。この脆弱性が悪用されると任意のコードが実行され、パソコンが不正に操作される危険性があります。この問題の影響を受ける製品と、その対策方法、回避策を以下にご案内いたしますので、ご確認の上、ご対応をお願いいたします。

脆弱性の内容

今回の脆弱性を悪用することを目的に改ざんされた文書ファイルを直接開いた場合、悪意のあるプログラムを実行しようとします。

[JS13003] 一太郎的漏洞利用惡意程式的執行所造成的風險

公開日：2013-11-12.

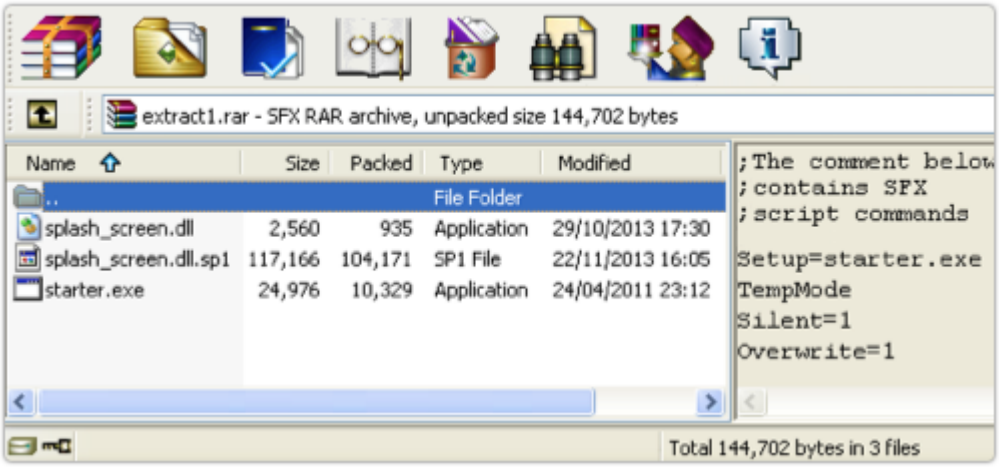
***概要：** 本公司的產品已確認了漏洞的存在。如果利用這個漏洞的惡意程式被執行，個人電腦將會有被惡意操控的危險性。受到這個影響的產品、其應對法、迴避方法如下，請您確認。*

***弱點內容：** 這個漏洞的惡意利用是在直接開啟受感染的文件檔時，將會啟動惡意程式。*

透過模擬該執行檔，可以發現它在文件內的 shellcode 會提取惡意程式的下一階段，我們可以看到接下來應該會發生什麼。

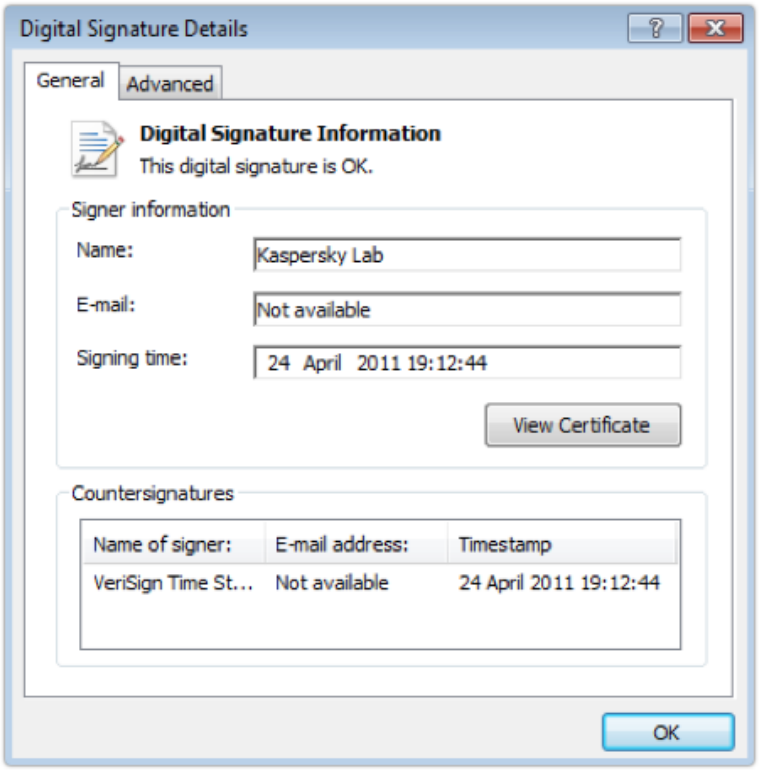
惡意程式如何載入

一個 WinRAR 自解檔，包含三個文件，寫入到磁碟並執行：



該自解檔將控制權交給 starter.exe，這是一個有老式的數位簽章，來自防毒廠商

卡巴斯基於 2011 年的認證的應用程式：



卡巴斯基元件被設計來載入一個名為 splash_screen.dll 的 DLL，但不會強制指定該 DLL 是從磁碟上的哪一個特定目錄裡載入。所以，如果恰好有一個假冒的 DLL 在 Windows 的 PATH（包括當前目錄）裡，假冒者將代替真正的 DLL 被執行。

→這就是所謂的‘不安全的程式庫載入漏洞’，它並且可以利用任何只需要 DLL 的名字的程式，例如：MY.DLL，而不是透過使用明確的路徑和文件名稱，如：C：\特定路徑\MY.DLL。給程式設計師的警告：在呼叫 LoadLibrary（）時，總是明確地指定 DLL 文件名。

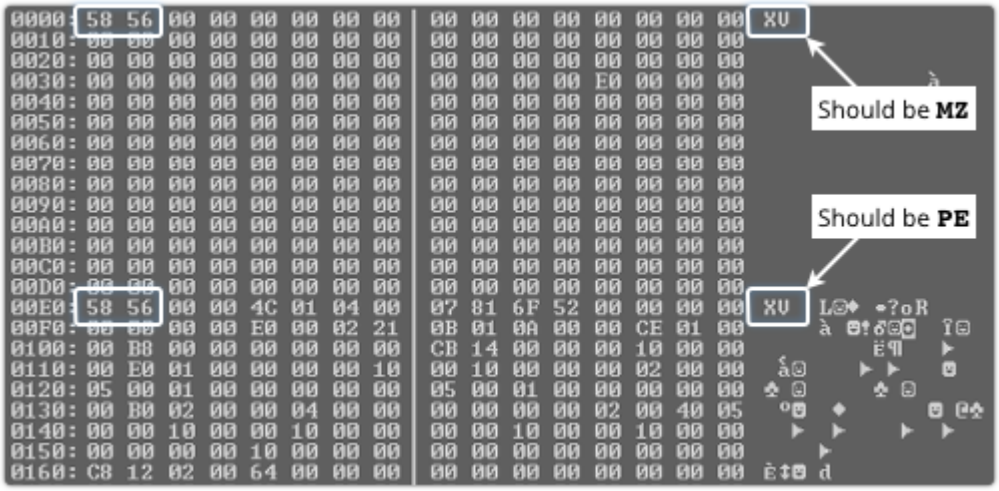
假冒的 DLL 讀取，解壓縮並載入惡意的文件，splash_screen.dll.sp1。

這裡是假冒的 DLL 呼叫在系統程式庫 NTDLL 中的 Windows 函數 RtlDecompressBuffer（）的程式碼：

```
                mov     [ebp+var_B0], 'ntdl'
                mov     [ebp+var_AC], 'l'
                call    [ebp+var_24]
                mov     esi, eax
                test    esi, esi
                jnz     short loc_379
                push    7
                jmp     error
; ---------------------------------------------------------------------------
loc_379:        lea     eax, [ebp+var_44]
                push    eax
                push    esi
                mov     [ebp+var_44], 'RtlD'
                mov     [ebp+var_40], 'ecom'
                mov     [ebp+var_3C], 'pres'
                mov     [ebp+var_38], 'sBuf'
                mov     [ebp+var_34], 'fer'
                call    edi
                mov     [ebp+var_C], eax
                test    eax, eax
                jnz     short loc_3B1
                push    8
                jmp     error
```

作為一個非常溫和的反分析技巧，解壓縮後的資料幾乎是一個正確的 Windows DLL 文件，但它並不完整。.

在下方圖的兩個 XV 實例，本來該分別為 MZ 和 PE，其中 MZ 表示該文件是微軟格式的可執行程式，和 PE 表示它是一個可移植可執行文件，或為 Windows 程式：



對假冒的 DLL 來說缺乏 MZ 和 PE 是無關的，因為它不依賴 Windows 載入解壓縮程序。

取而代之，假冒的 DLL 含有它自己的程式載入器，正確地執行 splash_screen.dll.sp1 的惡意內容程序到記憶體本身，然後跳到它上面。

當解壓縮內容後，假冒的 DLL 使用的標頭資訊（其中包括載入時間資料）如上所示，但一旦程式載入器準備跳進內容程序時，標頭資訊是多餘的。

因此，假冒的 DLL 在控制權轉交給內容本身之前以零覆寫了內容標頭。

藉由在形式上輕易的傾倒在磁碟上執行中的內容程序的不足記憶體，這將會立即的合理化微軟官方的 Windows 除錯工具，由此防止惡意程式分析師分析。

再一次的，這是一個非常溫和的反分析技巧。

→ 在實作中，你可以繞過這類的反分析手法透過使用偵錯工具停止假冒的 DLL，就在上面提到的呼叫 RtlDecompressBuffer（）後立即使用，與在它被抹消之前傾銷相關的標頭資料。

惡意程式如何工作

一個惡意程序已經載入到記憶體中，並準備使用，假冒的 DLL 跳到它上面。

程序連接到在 www 點 mofamails 點 com 的伺服器以取得有關下一步該怎麼做的指示。

如同許多的 BOT 與後門程式，這給了滲透者非常普遍的控制受感染的電腦，包括以下功能：

- 收集正在執行的程序和模組資訊。
- 載入和重新設定系統服務。
- 啟動和停止程序。
- 創建和刪除檔案。
- 操作登錄檔。
- 獲取詳細的系統資訊。
- 側錄鍵盤。
- 抓取截圖。
- 監視網絡資源和連結。

我們能學到什麼

經過這次攻擊所提出的明顯的問題是：為什麼是日本？為什麼是一太郎？

一個答案是「我們不知道。」另外一個答案，令人難過，是「為什麼不？」

據我們所知，該 PlugX 作者獲得的一太郎漏洞，給他們一個方式來攻擊他們以前打不到的受害者，所以他們決定嘗試一下。

這提醒我們的是，我們都有被漏洞攻擊的潛在風險，也不只我們這些使用主流作業系統和產品的市場。（Apple 用戶留意！）

簡而言之：

- 不要打開你沒有預料到會寄來的附件，不管他們看起來有多可信。
- 確保使用的所有軟體都更新到了最新版。
- 使用及時的防毒軟體，並確保它更新到最新版。