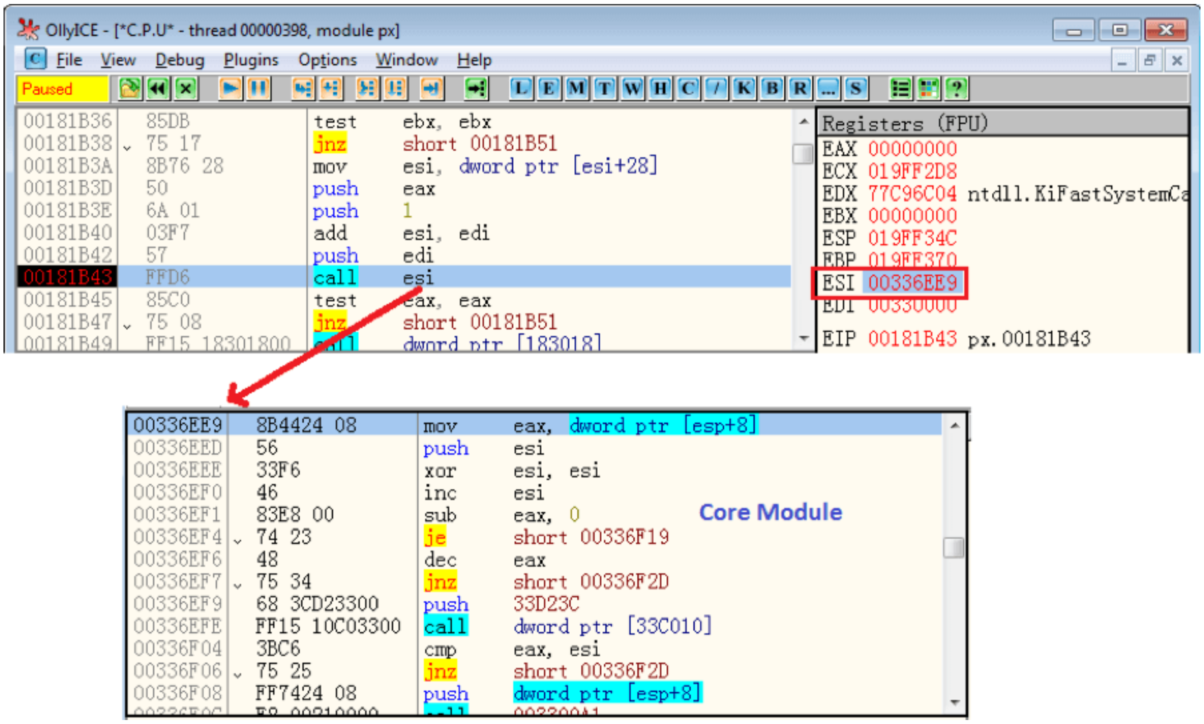
New Variant of Ursnif Continuously Targeting Italy

fortinet.com/blog/threat-research/new-variant-of-ursnif-continuously-targeting-italy

January 12, 2021


OllyICE - [*C.P.U* - thread 00000398, module px]
File View Debug Plugins Options Window Help
Paused
L E M T W H C / K B R ... S
Registers (FPU)
Core Module

```
00181B36    85DB            test    ebx, ebx
00181B38  ⌄ 75 17           jnz     short 00181B51
00181B3A    8B76 28         mov     esi, dword ptr [esi+28]
00181B3D    50              push    eax
00181B3E    6A 01           push    1
00181B40    03F7            add     esi, edi
00181B42    57              push    edi
00181B43    FFD6            call    esi
00181B45    85C0            test    eax, eax
00181B47  ⌄ 75 08           jnz     short 00181B51
00181B49    FF15 18301800   call    dword ptr [183018]
```

```
EAX 00000000
ECX 019FF2D8
EDX 77C96C04 ntdll.KiFastSystemCa
EBX 00000000
ESP 019FF34C
EBP 019FF370
ESI 00336EE9
EDI 00330000

EIP 00181B43 px.00181B43
```

```
00336EE9    8B4424 08       mov     eax, dword ptr [esp+8]
00336EED    56              push    esi
00336EEE    33F6            xor     esi, esi
00336EF0    46              inc     esi
00336EF1    83E8 00         sub     eax, 0
00336EF4  ⌄ 74 23           je      short 00336F19
00336EF6    48              dec     eax
00336EF7  ⌄ 75 34           jnz     short 00336F2D
00336EF9    68 3CD23300     push    33D23C
00336EFE    FF15 10C03300   call    dword ptr [33C010]
00336F04    3BC6            cmp     eax, esi
00336F06  ⌄ 75 25           jnz     short 00336F2D
00336F08    FF7424 08       push    dword ptr [esp+8]
```

FortiGuard Labs Threat Research Report

Affected platforms:	Microsoft Windows
Impacted parties:	Windows Users in Italy
Impact:	Collects Victims' Information
Severity level:	Critical

Ursnif (also known as Gozi) is identified as a banking Trojan, but its variants also include components (backdoors, spyware, file injectors, etc.) capable of a wide variety of behaviors.

The Ursnif Trojan has been observed targeting Italy over the past year. A few days ago, FortiGuard Labs detected a phishing campaign in the wild that was spreading a fresh variant of the Ursnif Trojan via an attached MS Word document that is continuously targeting Italy.

Although Ursnif is identified as a banking Trojan, due to its C2 server's shutdown, this latest variant has been unable download the malicious banking module it needs to steal banking information from the victim, causing it to fail to start the second stage of its attack. As a result, in this post I will share my findings around the first stage of this campaign. You will learn what the phishing email looks like, how the MS Word document attached to the email works to download Ursnif, as well as what this variant does on a victim's device.

Ursnif Phishing Email

Figure 1.1 is a screenshot of the Ursnif phishing email. As you can see, it was written in Italian and masquerades as a payment reminder.

Figure 1.1. Phishing Email and attached MS Word document

I used Google Translate to translate the email content into English:

Dear customer,

A recent accounting audit shows that your invoice number 294316 of 12/10/2020 expired on 12/11/2020. As of today, it is not yet been paid by you.

Therefore, please normalize your accounting position as soon as possible. We are also reminding you that this payment can be made by bank transfer using the IBAN indicated in the invoice or, by bank check or bank draft.

You can consult the invoice and the details for the payment through the attached archive.

We thank you for your attention and we send you kind regards.

Attached to the email is an MS Word document named “residuo_8205843.doc”. The text lures the victim into opening the document to get more details of the invoice.

Word Document Analysis

As you may have guessed, the Word document contains malicious Macros. Once the victim opens the document in MS Word, it pops up a yellow warning bar to alert the user that the file contains Macros, as shown in Figure 2.1.

Figure 2.1. Opening the attached document in MS Word program

Once the victim clicks the button to enable the Macros, malicious Macro code will be executed in background. A built-in function, “Document_Open()”, is automatically called first when the document is opened, as shown in Figure 2.2.

Figure 2.2. Built-in function Document_Open()

I noticed that the attacker creates a hidden UserForm (UserForm1) in the project, and a button’s click event function is invoked from the Document_Open() function.

Figure 2.3. Hidden UserForm

The main work of the Macro is implemented using this form. In Figure 2.2, we can see that it calls “UserForm1.CommandButton2_Click”, which belongs to the event function of the “Process” button, as shown in Figure 2.3. It functions the same as when the victim clicks the “Process” button. Other clicked buttons also directly call their event functions in code.

The Macro then downloads a DLL file from a hardcoded URL, “longline[.]cyou/p1cture3[.]jpg” (refer to Figure 2.4) and resaves it into the file "C:\users\public\px.dat". The final step the Macro performs is to run the downloaded DLL using the Windows program “RegSvr32.exe”. Figure 2.4 is the screenshot of when the Macro runs this DLL.

Figure 2.4. Downloading and Running a DLL file from the hardcoded URL

Analyzing the Downloaded DLL File

Like most other malware protected by a packer, this downloaded DLL file (C:\users\public\px.dat) is similarly protected by a packer as well. It is started by RegSvr32.exe, and the unpack program gets called to first extract Ursnif into the memory. Ursnif’s DllEntryPoint() function will be called at the end by the unpack program. Figure 3.1 is a screenshot of the unpacked DllEntryPoint() function.

Figure 3.1. Unpacked DllEntryPoint() function

According to my analysis, this function’s main task is to decompress another PE file that is secretly kept in the “.reloc” section. Next, it loads this PE file into a file mapping memory by calling API CreateFileMappingW() and MapViewOfFile()) and executing the code in it. This decompressed PE file is the core module of this variant of Ursnif.

Figure 3.2. Call core module’s entry function

In Figure 3.2, we can see that the malware is about to call the entry function of the core module that has been loaded at the base address 0x330000.

Decrypting the Configuration Block in the Core Module

Ursnif has an encrypted configuration block inside the PE’s “.bss” section to protect its data from easy analysis. So, decrypting the configuration block is the first thing that it needs to do. The code snippet responsible for decrypting the configuration block is shown in Figure 4.1, below.

Figure 4.1. Code snippet for decrypting the configuration block

Going through the decrypted configuration block, we can see that it contains a number of ASCII and Unicode strings, as well as binary data. The values from the configuration block are accessed throughout the core module. Let’s see what it can get from the decrypted configuration block. Following is a partial list of its strings.

"invalidcert"

"overridelink"

"%08X-%04X-%04X-%04X-%08X%04X"

"StdRegProv"

"/images/"

"version=%u&soft=%u&user=%08x%08x%08x%08x&server=%u&id=%u&type=%u&"

"name=%s"

"Content-Disposition: form-data; name="upload_file"; filename="%s""

"DeleteKey"

"Mozilla/4.0 (compatible; MSIE 8.0; Windows NT %u.%u%s)"

"soft=%u&version=%u&user=%08x%08x%08x%08x&server=%u&id=%u&crc=%x"

"&uptime=%u"

"CreateProcessA"

".avi"

"Software\Microsoft\Windows\CurrentVersion\Run"

"rundll32 "%s",%S"

"&ip=%s"

"&os=%s"

"&tor=1"

"&dns=%s"

"&whoami=%s"

"/C "copy "%s" "%s" /y && rundll32 "%s",%S""

"SOFTWARE\Microsoft\Windows NT\CurrentVersion"

"WScript.Shell"

");%S.Run("

"powershell iex ([System.Text.Encoding]::ASCII.GetString(("

"IE8RunOnceLastShown_TIMESTAMP"

"SOFTWARE\Microsoft\Internet Explorer\"

…

Collecting Sensitive Information and Sending It to the C2 Server

In the core module, Ursnif collects sensitive information from the victim’s device, such as current login Username, Computer Name, System Uptime, and so on. These are formatted as in the first packets sent to the C2 server. Figure 5.1 shows the ASM code snippet used to obtain Username and Computer Name.

Figure 5.1. Calling APIs to obtain Username and Computer Name

It formats all of the collected data into key=value pairs, like the string shown in the memory section of Figure 5.2.

Figure 5.2. Collected data in a formatted string

Among the key-value pairs, the values of “soft”, “version”, and “crc” are hardcoded. The values of “user”, “server”, and “id” are hash values. “uptime” is a time value of how long the device has been running. “dns” is the computer name, and “whoami” is the full user name.

Next, Ursnif encrypts the string and encodes the cipher text with base64. Finally, it transforms the base64 string by replacing specified bytes and randomly inserting “/”. An example looks like this:

kGmre5fgzV/jtkp5rBSbtChvz3cy/avjHTlxQBzmZ/DH4YpgdvFWT/3ZOkljxvOGl5sK/DKS8qy7zKWSwKdmWKVkNW/3iLRdsyjEatQ2kbQ/97eIek932
TCOCQKSzO32C/EUgtTKuNa/PDAfPobZLDuYwMNjMJqk/BCloHCcWrST2Oq7lSc7/bX0t3dF5oQwhDoo8O83JoQ/KNQw3uRXGoOK3/bvb2q8hp/
PoIH_2/F

It then adds a prefix string (the domain of the C2 server and “/images/”) and suffix string (“.avi”) before sending the information to the server.

Ursnif does not directly send the data within the process. Instead, it uses a COM component—"IE ActiveX Interface"—that is implemented in the module "IEProxy.dll". Ursnif indirectly calls APIs from this module to perform communication with C2 server. This means the collected data will be sent by a newly created "iexplorer.exe" process that is started by the COM component. Figure 5.3 provides the detailed overview processes tree, from which you can not only observe the process "iexplorer.exe", but also the relationship between the processes mentioned in this blog, such as "WinWord.exe" and "RegSvr32.exe".

Figure 5.3. Process Tree of calling a COM component

Figure 5.4. Decrypted C2 server domain string

The domain string of the C2 server comes from a secret data structure that is decrypted from a data block within the ".reloc" section.

In Figure 5.4, at the bottom of the memory segment, we can see the decrypted secret data structure. The string starting at offset 0xB0 are the domains: "web[.]vortex[.]data[.]Microsoft[.]com", "ocsp[.]sca1b[.]amazontrust[.]com", and "gstatistics[.]co". Through my analysis, only the last one is the real C2 server domain.

Figure 5.5 shows what the final packet sent to the C2 server looks like.

Figure 5.5. Packet sent to the C2 server

In the next stage, the malware would respond to the received packet with a module (dll, exe, etc.) which would be executed by Ursnif to perform further malicious attacks on victim's device. Unfortunately, we did not receive any response because the C2 server had been shut down. I will continue to try and contact the C2 server. If I am finally able to obtain the module file, I will post my analysis of it as well.

Fortinet Solution Coverage:

The Word document attached to the phishing email has been detected as "**VBA/Ursinf.3412!tr**" and the downloaded file has been detected as "**W32/Ursinf.KB!tr**" by the FortiGuard AntiVirus service.

The URL used to download Ursnif (DLL file) has been rated as "**Malicious Websites**" by the FortiGuard WebFilter service.

The CDR (Content Disarm & Reconstruction) feature can also neutralize this threat by removing all malicious Macro code.

The FortiGuard AntiVirus service is supported by FortiGate, FortiMail, FortiClient and FortiEDR. And the CDR feature is supported by FortiGate and FortiMail.

We also suggest our readers to go through the free NSE training -- NSE 1 – Information Security Awareness, which has a module on Internet threats designed to help end users learn how to identify and protect themselves from phishing attacks.

IoCs:

URLs:

"hxxp://longline[.]cyou/p1cture3[.]jpg "

"hxxps://gstatistics[.]co/"

Sample SHA256:

residuo_8205843.doc:

E9732CDCA1B2503E02E8FEA9A4C68EDA940E10890E1C5ABE2CEB2290FE39C3DB

Downloaded DLL file (px.dat or p1cture3.jpg):

90D8648B2AAC0C837286A4C042F02064CFBB12F45B3DC6B00B2BECCC7FC35422

Learn more about FortiGuard Labs threat research and the FortiGuard Security Subscriptions and Services portfolio.

Learn more about Fortinet's free cybersecurity training initiative or about the Fortinet NSE Training program, Security Academy program, and Veterans program.