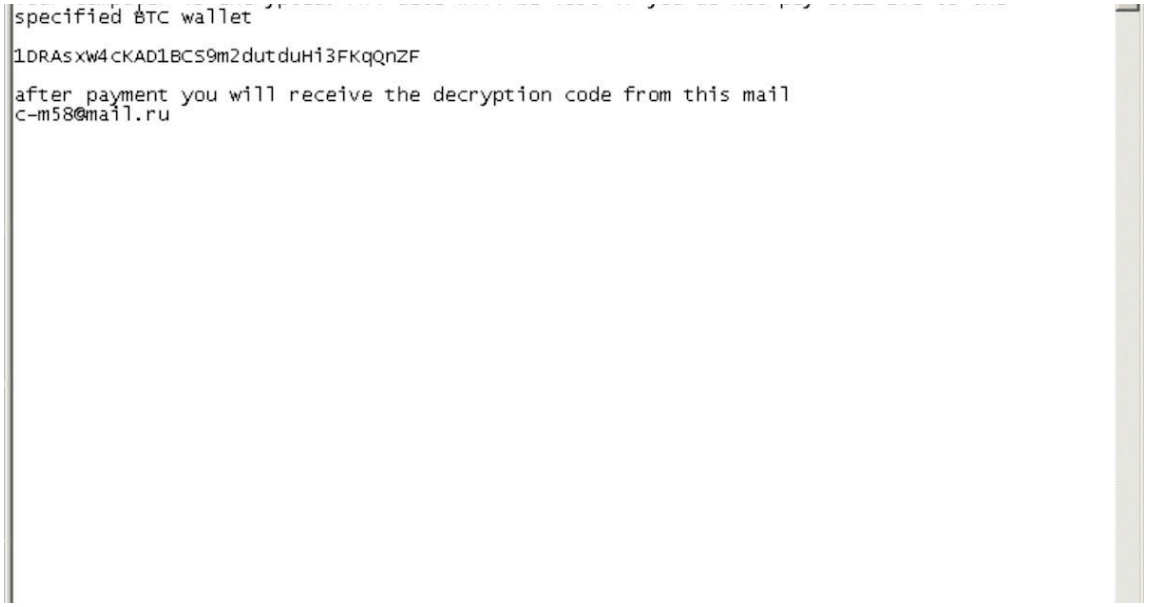
Files Cannot Be Decrypted? Challenge Accepted. Talos Releases ThanatosDecryptor

By Edmund Brumaghin

Published: 2018-06-26 · Archived: 2026-04-05 19:14:02 UTC



Tuesday, June 26, 2018 11:00

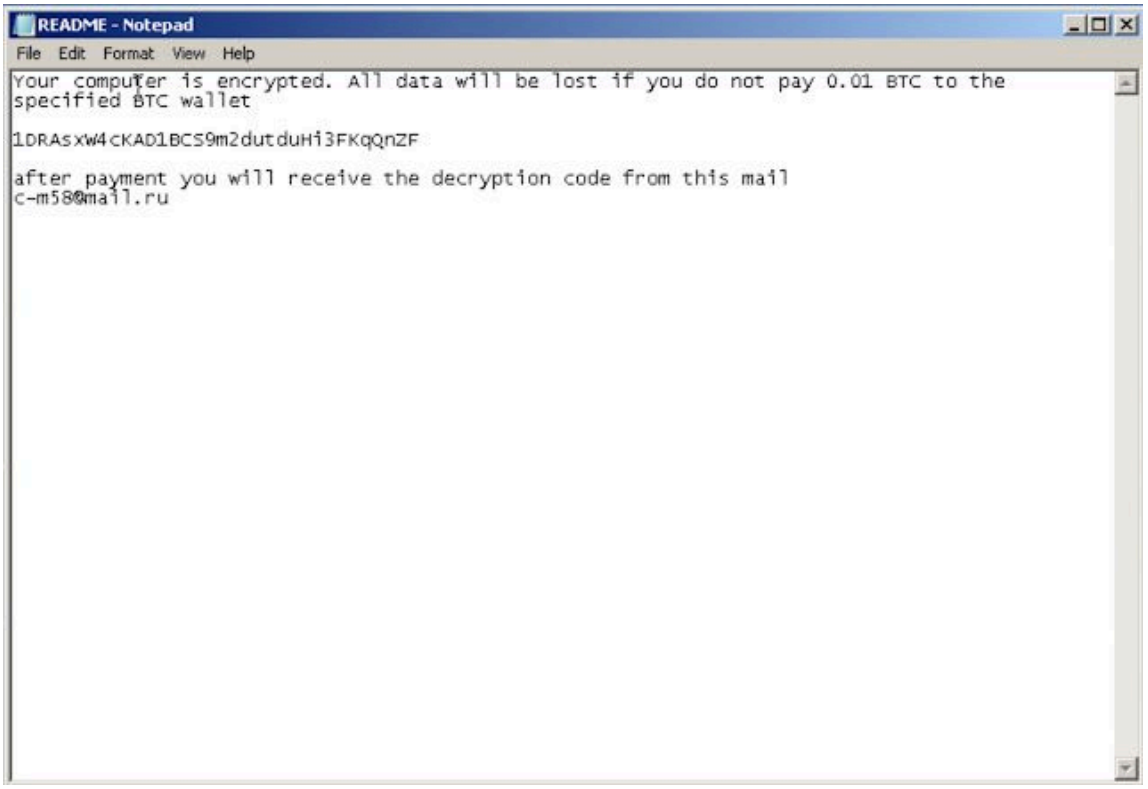
This blog post was authored by Edmund Brumaghin, Earl Carter and Andrew Williams.

Additionally, due to issues present within the encryption process leveraged by this ransomware, the malware authors are unable to return the data to the victim, even if he or she pays the ransom. While previous reports seem to indicate this is accidental, specific campaigns appear to demonstrate that in some cases, this is intentional on the part of the distributor. In response to this threat, Talos is releasing ThanatosDecryptor, a free decryption tool that exploits weaknesses in the design of the file encryption methodology used by Thanatos. This utility can be used by victims to regain access to their data if infected by this ransomware.

Technical details

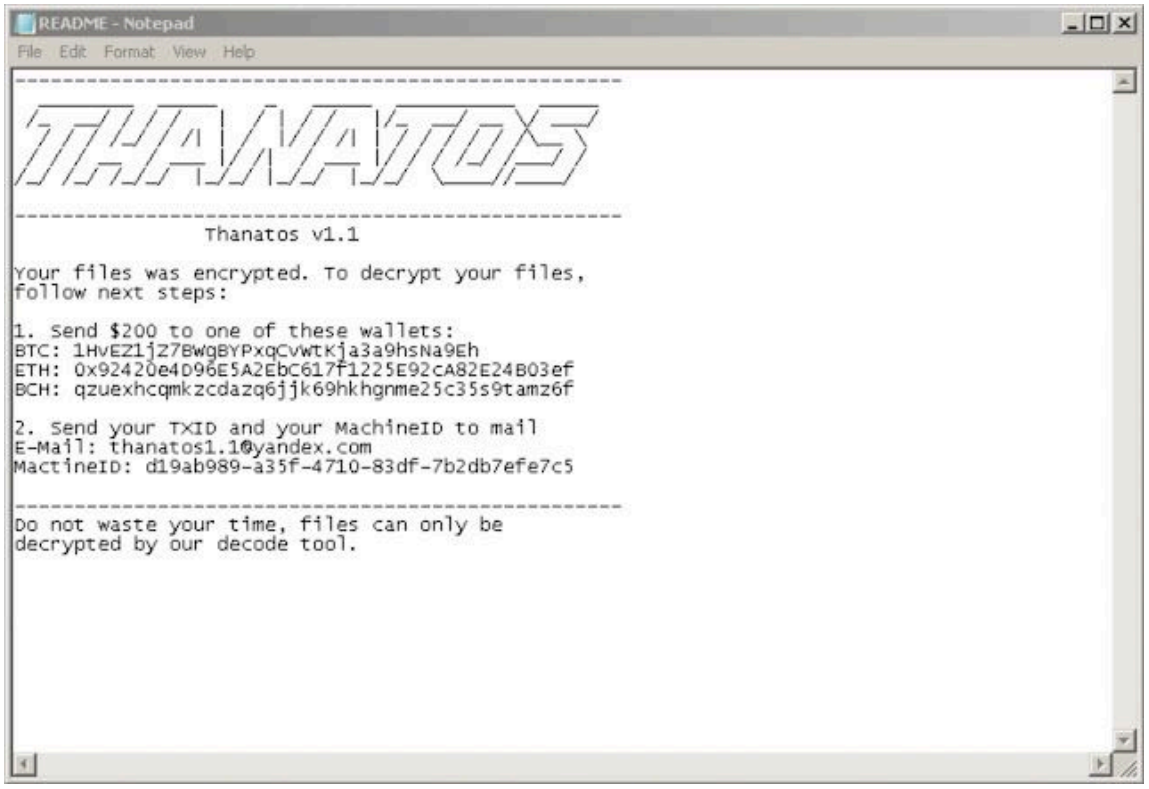
Ongoing evolution of Thanatos

While tracking and analyzing the various campaigns being used to distribute the Thanatos ransomware, Talos identified multiple distinct versions of this malware, indicating that it is continuing to be actively developed by the malware author. The main differences can be directly observed within the ransom note being used to inform victims that they have been infected and provide instructions for paying a ransom to the attacker. Version 1 of Thanatos, which was being distributed in mid-February of this year, featured a very primitive ransom note that is stored on the victim's desktop as README.txt.



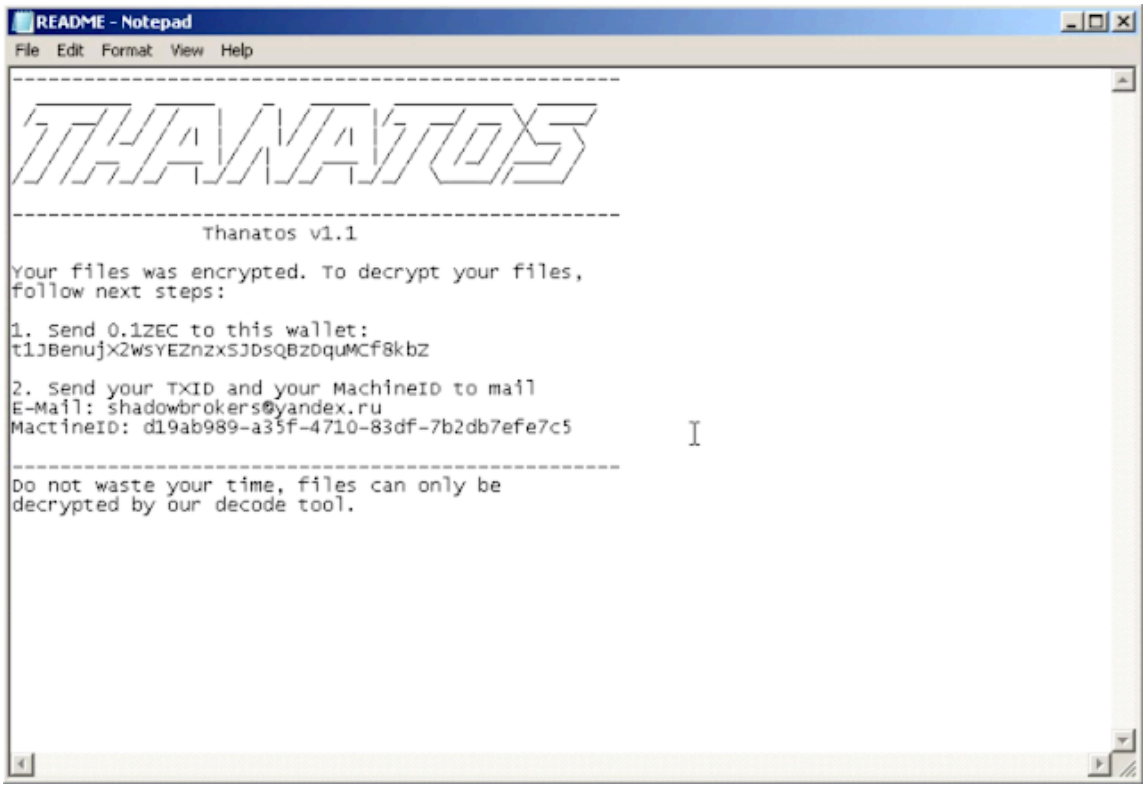
In this version of Thanatos, the ransom note simply informs the user that their files have been encrypted and instructs them to pay a ransom amount of 0.01 bitcoin (BTC) to the specified bitcoin wallet. Rather than using different wallet addresses across samples, the same hardcoded wallet address is present in all samples of this version of Thanatos that Talos analyzed. Payment processing appears to be manual and email-based, which is indicative of an attacker with limited resources and knowledge of ransomware creation and distribution techniques used by other more well-known ransomware families such as Locky, Cerber, etc.

Shortly after Version 1 was observed being distributed, malware distribution campaigns began distributing Thanatos Version 1.1 with the majority of the distribution of Version 1.1 occurring between February and April 2018. This updated version of Thanatos featured several key differences related to the type of cryptocurrencies that victims could pay with.



As can be seen in the screenshot of the ransom note above, Thanatos Version 1.1 supports payment of the ransom demand using BTC, ETH, and BCH. Additionally, the malware also now includes a unique MachineID that the victim is instructed to send to the attacker via email.

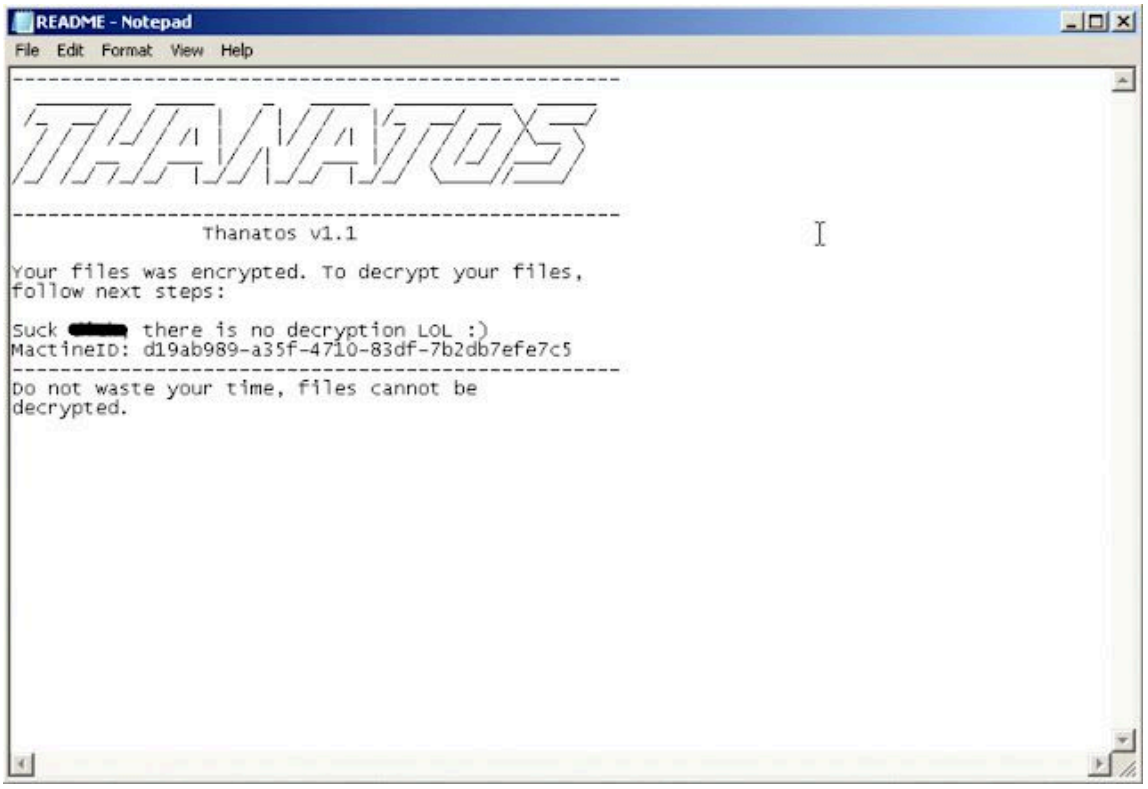
Interestingly, the ransom notes changed several times across samples that Talos analyzed. Below is another example of one of the ransom notes used by this malware. Note that the attacker had changed the email address being used to communicate with victims. The attacker was also purporting to process ransom payments in the form of Zcash versus the other cryptocurrencies listed in the other ransom notes.



In investigating the distribution mechanisms being used by the attacker to infect victims and remove their ability to access data on their system, we identified an interesting campaign that indicated that at least in this particular case, the attacker had no intention of providing any sort of data decryption to the victim. The malware appears to have been delivered to the victim as an attachment to a chat message sent to the victim using the Discord chat platform. Discord is a voice and text chatting platform that allows direct communications between two or more participants. The URL hosting the attached malware is below:

hxxps://cdn[.]discordapp[.]com/attachments/230687913581477889/424941165339475968/fastleafdecay.exe

The filename used in this case was "fastleafdecay.exe" which may indicate that the victim was tricked into executing the malware as it was posing as a mod of the same name in the video game Minecraft. When executed, this sample displayed the following ransom note to victims:



As can be seen in the above screenshot, the malware author did not include any instructions for paying a ransom, instead stating that decryption was not available, indicating that this particular case was not financially motivated, and instead was used to destroy data on the victim's system. Interestingly, the PDB path that was intact on this sample differed from the other samples that Talos analyzed. In this case, the PDB path was:

C:\Users\Artur\Desktop\csharp - js\кости пизда\Release\Thanatos.pdb

Most of the other samples contained the following PDB path:

D:\Work\Thanatos\Release\Thanatos.pdb

Talos also observed a sample that had been compiled in debug mode that contained the following PDB path:

D:\Работа\Локер шифровчик\Thanatos-master\Debug\Thanatos.pdb

Thanatos operations and encryption process

When executed on victim systems, Thanatos copies itself into a subdirectory that it creates within %APPDATA%/Roaming. The subdirectory name and executable file name are randomly generated based on system uptime and changes each time the malware executes.

Thanatos recursively scans the following directories within the current user's profile to identify files to encrypt:

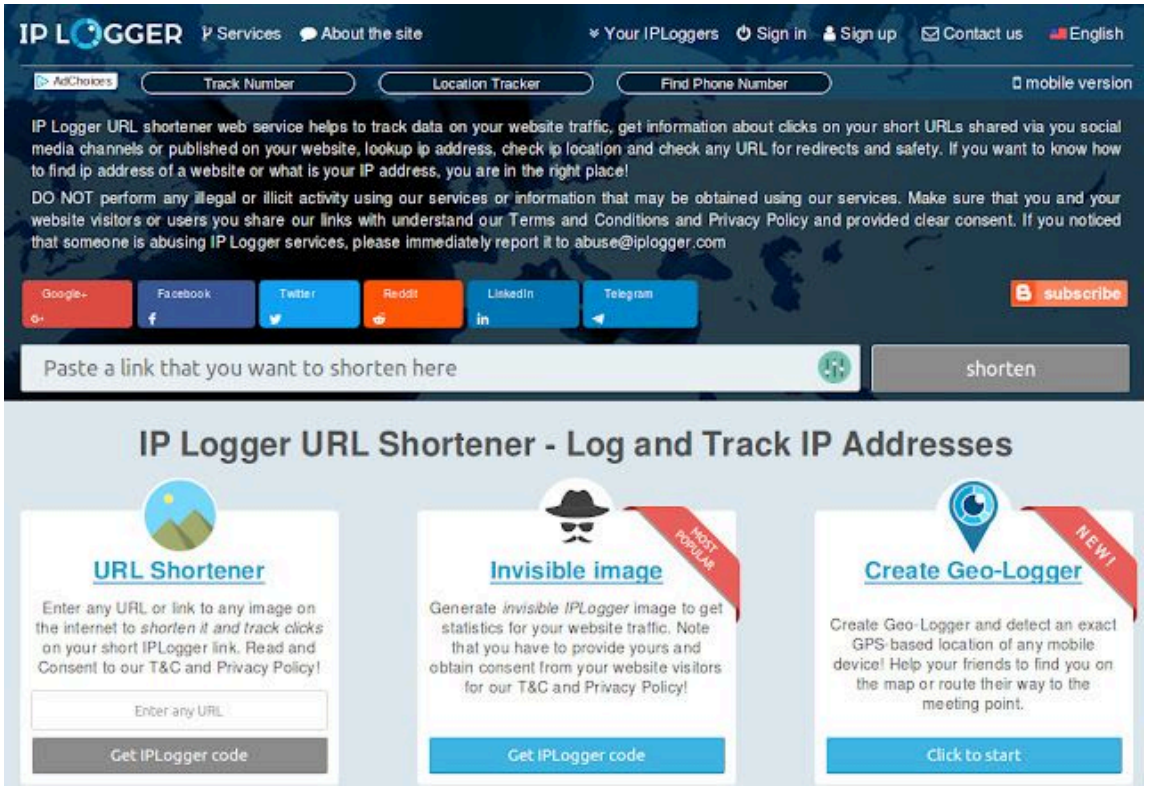
Desktop
- Documents
- Downloads
- Favourites
- Music

OneDrive

Pictures
Videos

While many ransomware families have a specific list of file extensions that are supported for encryption, Thanatos supports encryption of any file that has an extension. For each file that the malware locates, it derives an encryption key based on the number of milliseconds that the infected system has been running via a call to GetTickCount. The malware then encrypts the file using Advanced Encryption Standard (AES)-256 and discards the encryption key. The process of discarding the encryption key precludes the attacker from being able to provide access to the decrypted data, even if a ransom demand is paid. Encrypted files are then written to the filesystem with the .THANATOS file extension and the original files are deleted.

The malware also leverages an external website called iplogger. This website provides customized URLs that can be used to track information about systems that access the URL. By making HTTP GET requests using these hardcoded URLs, the attacker can obtain information about all of the different systems that have been infected with Thanatos.



The HTTP GET request are all made using the following user agent:

Mozilla/5.0 (Windows NT 6.1) Thanatos/1.1

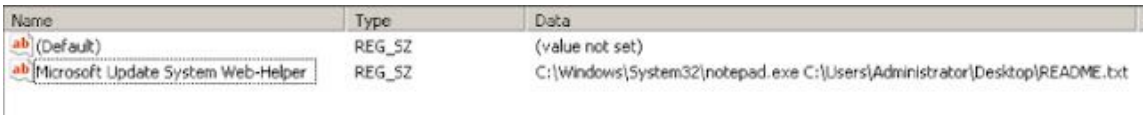
```
GET /1t3i37 HTTP/1.1
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 6.1) Thanatos/1.1
Host: iplogger.com

HTTP/1.1 301 Moved Permanently
Server: nginx
Date: Wed, 06 Jun 2018 17:59:19 GMT
Content-Type: text/html
Content-Length: 178
Connection: keep-alive
Location: https://iplogger.com/1t3i37
Expires: Thu, 31 Dec 2037 23:55:55 GMT
Cache-Control: max-age=315360000

<html>
<head><title>301 Moved Permanently</title></head>
<body bgcolor="white">
<center><h1>301 Moved Permanently</h1></center>
<hr><center>nginx</center>
</body>
</html>
```

Talos has observed the following iplogger URLs hardcoded into various Thanatos samples that were analyzed:

hxxp://iplogger[.]com:80/1CUTM6
hxxp://iplogger[.]com:80/1t3i37

The ransom note associated with Thanatos is saved to the infected user's desktop using the filename README.txt. A registry entry is created so that each time the system boots, the ransom note is displayed using the Notepad application. This registry key is located in:

HKCU\Software\Microsoft\Windows\CurrentVersion\Run



Aside from this, the malware does not obtain persistence for the executable itself.

ThanatosDecryptor

As previously described, the encryption keys used to encrypt files on victims' systems are derived based upon the number of milliseconds since the system last booted. This value is a 32-bit number, meaning that the encryption key is effectively 32 bits as well. Additionally, the maximum number of milliseconds that can be stored in a 32-bit value is roughly 49.7 days' worth, which is higher than the average amount of uptime on many systems due to patch installation, system reboots, and other factors. This makes brute-forcing the key values significantly cheaper from a time perspective.

Another optimization can be made based on the fact that the system uptime is written to the Windows Event Log roughly once per day. Since Thanatos does not modify the file creation dates on encrypted files, the key search space

can be further reduced to approximately the number of milliseconds within the 24-hour period leading up to the infection. At an average of 100,000 brute-force attempts per second (which was the baseline in a virtual machine used for testing), it would take roughly 14 minutes to successfully recover the encryption key in these conditions.

Talos is releasing a decryption utility that can be leveraged by victims of Thanatos to attempt to regain access to data and files stored on the infected system. It has been tested on Versions 1 and 1.1 of the Thanatos ransomware and on all currently known Thanatos samples Talos has observed.

Note: In order to decrypt files as quickly as possible, ThanatosDecryptor should be executed on the original machine that was infected and against the original encrypted files that the malware created.

This decryption utility currently supports decryption of the following types of files:

Image: .gif, .tif, .tiff, .jpg, .jpeg, .png
Video: .mpg, .mpeg, .mp4, .avi
Audio: .wav
Document: .doc, .docx, .xls, .xlsx, .ppt, .pptx, .pdf, .odt, .ods, .odp, .rtf
Other: .zip, .7z, .vmdk, .psd, .lnk

The decryptor first searches the same directories as the ransomware to identify files that contain the .THANATOS file extension. For files that contain the .THANATOS file extension, the decryptor will then obtain the original file extension, which is left intact during infection, and compare it to the list of supported file types. If the file type is supported, the decryptor will then queue that file for decryption.

ThanatosDecryptor also parses the Windows Event Log for uptime messages and uses the encrypted file creation time metadata to determine a starting value for decryption. This value is used to derive an encryption key, and an AES decryption operation is performed against the file contents. The resulting bytes are then compared against values known to be valid file headers for the specific file type. If they do not match, meaning the decryption process was unsuccessful, the seed value for the encryption key is then incremented, and the process is repeated. Once successful, the original file is written to the file system, and the original filename is restored. Once one file has been successfully decrypted, ThanatosDecryptor uses the seed value from that decryption attempt as the starting point for decryption attempts against additional files since they are likely to be very similar.

To execute ThanatosDecryptor, simply download the ThanatosDecryptor project here and execute ThanatosDecryptor.exe, which can be found in the release directory. Additional information and example output can be obtained here.

Following the money … or lack thereof

As previously mentioned, throughout the various Thanatos campaigns and associated samples, the attacker behind this threat made changes to the types of cryptocurrencies that they claim are supported for paying the ransom demand. Analysis of these various wallets and associated cryptocurrency transactions revealed some interesting information about the size and success of these malware campaigns over time. Across all of the samples, the following cryptocurrency wallets were listed along with instructions for paying the ransom on the ransom note accompanying the malware.

Bitcoin ($BTC):
1HVEZ1jZ7BWgBYPxqCVWtKja3a9hsNa9Eh

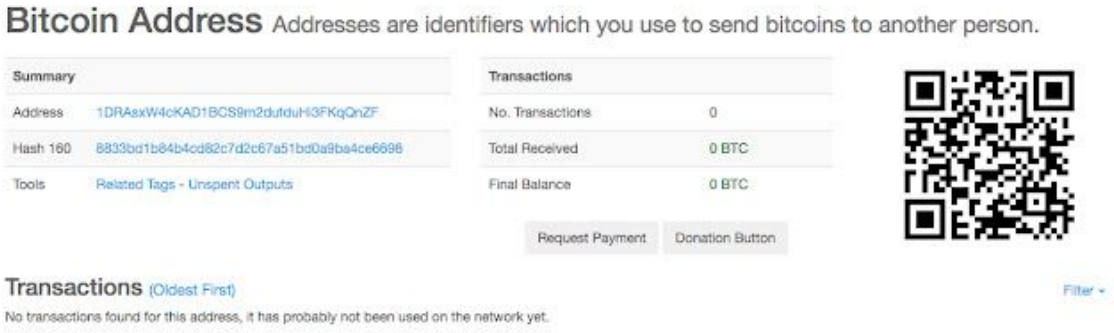
1DRAsxW4cKAD1BCS9m2dutduHi3FKqQnZF

Ethereum ($ETH):
0x92420e4D96E5A2EbC617f1225E92cA82E24B03ef

Bitcoin Cash ($BCH):
Qzuexhcqmkzcdazq6jjk69hkhgnme25c35s9tamz6f

ZCash ($ZEC):
t1JBenujX2WsYEZnzxSJDsQBzDquMCf8kbZ

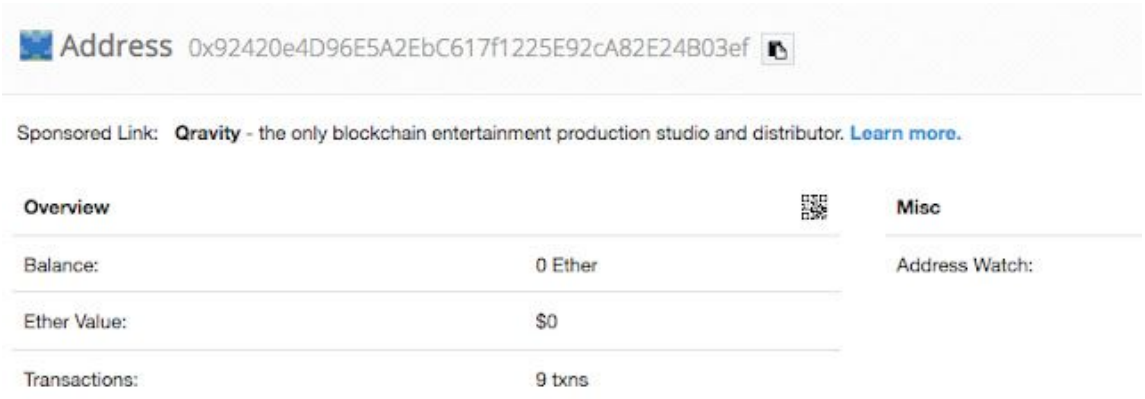
In analyzing the bitcoin wallets, we identified that the attacker had not received a single ransom payment from victims. In fact, the wallet listed most frequently across the samples analyzed (1HVEZ1jZ7BWgBYPxqCVWtKja3a9hsNa9Eh) was not even a valid bitcoin wallet. This means that even if a victim tried to pay using bitcoin, they would have been unable to. The second wallet (1DRAsxW4cKAD1BCS9m2dutduHi3FKqQnZF) did not have a single transaction to or from it.



Likewise, the Bitcoin Cash wallet that was listed has also never seen a single transaction.

When analyzing the Zcash wallet that was seen listed on one of the ransom notes associated with Thanatos, we identified that while it had seen several transactions, the total amount of ZEC received by this wallet was 2.24767084, which equals approximately $450 USD.

Finally, the Ethereum wallet used by the attacker also saw several transactions. However, the total amount was also low compared to some of the more successful ransomware campaigns we regularly observe across the threat landscape. The total amount of ETH received in this wallet was 0.52087597, which equals approximately $270 USD.



This means that across all of the samples seen in the wild, the attacker's wallets had only received a total of $720 USD. If the incoming cryptocurrency was directly related to victims paying a ransom as a result of Thanatos infections, this clearly did not generate significant revenue for the attacker when compared to other financially motivated cybercrime operations.

Conclusion

Whether for monetary gains or to destroy data, attackers are continuously targeting end users. This malware proves how easy it has become for anyone to target users. You do not have to be a sophisticated attacker to cause havoc. There are also an endless supply of attack vectors available. In this case, for instance, the attacker took advantage of the Discord chat platform. Therefore, it is important to take security seriously and take steps to secure your systems, whether they are used for personal or business purposes. Since many of these attacks take advantage of users, you also need to be careful when opening attachments from unknown sources or clicking on unknown links.

Coverage

Additional ways our customers can detect and block this threat are listed below.

PRODUCT	PROTECTION
AMP	✔
CloudLock	N/A
CWS	✔
Email Security	✔
Network Security	✔
Threat Grid	✔
Umbrella	✔
WSA	✔

Advanced Malware Protection (AMP) is ideally suited to prevent the execution of the malware used by these threat actors.

Cisco Cloud Web Security (CWS) or Web Security Appliance (WSA) web scanning prevents access to malicious websites and detects malware used in these attacks.

Email Security can block malicious emails sent by threat actors as part of their campaign.

Network Security appliances such as Next-Generation Firewall (NGFW), Next-Generation Intrusion Prevention System (NGIPS), and Meraki MX can detect malicious activity associated with this threat.

AMP Threat Grid helps identify malicious binaries and build protection into all Cisco Security products.

Umbrella, our secure internet gateway (SIG), blocks users from connecting to malicious domains, IPs, and URLs, whether users are on or off the corporate 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.

YARA Signatures

Talos is also providing the following YARA signature that can be used to identify samples associated with the Thanatos ransomware family.

```
 rule Thanatos

{
        strings:

        $s1 = ".THANATOS\x00" ascii
        $s2 = "\\Desktop\\README.txt" ascii
        $s3 = "C:\\Windows\\System32\\notepad.exe C:\\Users\\" ascii
        $s4 = "AppData\\Roaming" ascii
        $s5 = "\\Desktop\x00" ascii
        $s6 = "\\Favourites\x00" ascii
```

```
        $s7 = "\\OneDrive\x00" ascii
        $s8 = "\\x00.exe\x00" ascii
        $s9 = "/c taskkill /im" ascii
        $s10 = "Software\\Microsoft\\Windows\\CurrentVersion\\Run" ascii

        condition:
        6 of ($s1, $s2, $s3, $s4, $s5, $s6, $s7, $s8, $s9, $s10)
}
```

Indicators of Compromise (IOC)

File Hashes (SHA256)

bad7b8d2086ac934c01d3d59af4d70450b0c08a24bc384ec61f40e25b7fbfeb5
fe1eafb8e31a84c14ad5638d5fd15ab18505efe4f1becaa36eb0c1d75cd1d5a9
8df0cb230eeb16ffa70c984ece6b7445a5e2287a55d24e72796e63d96fc5d401
97d4145285c80d757229228d13897820d0dc79ab7aa3624f40310098c167ae7e
55aa55229ea26121048b8c5f63a8b6921f134d425fba1eabd754281ca6466b70
02b9e3f24c84fdb8ab67985400056e436b18e5f946549ef534a364dff4a84085
241f67ece26c9e6047bb1a9fc60bf7c45a23ea1a2bb08a1617a385c71d008d79
0bea985f6c0876f1c3f9967d96abd2a6c739de910e7d7025ae271981e9493204
42748e1504f668977c0a0b6ac285b9f2935334c0400d0a1df91673c8e3761312

URLs

hXXps://cdn[.]discordapp[.]com/attachments/230687913581477889/424941165339475968/fastleafdecay.exe
hXXp://iplogger[.]com:80/1CUTM6
hXXp://iplogger[.]com:80/1t3i37

User Agents

Mozilla/5.0 (Windows NT 6.1) Thanatos/1.1

Source: https://blog.talosintelligence.com/2018/06/ThanatosDecryptor.html