Talos: a prototype Intrusion Detection and Prevention system for profiling ransomware behaviour

Ashley Charles Wood, Thaddeus Eze, Lee Speakman
University of Chester, Chester, United Kingdom
ashley.wood@chester.ac.uk
t.eze@chester.ac.uk
l.speakman@chester.ac.uk

Abstract: In this paper, we profile the behaviour and functionality of multiple recent variants of WannaCry and CrySiS/Dharma, through static and dynamic malware analysis. We then analyse and detail the commonly occurring behavioural features of ransomware. These features are utilised to develop a prototype Intrusion Detection and Prevention System (IDPS) named Talos, which comprises of several detection mechanisms/components. Benchmarking is later performed to test and validate the performance of the proposed Talos IDPS system and the results discussed in detail. It is established that the Talos system can successfully detect all ransomware variants tested, in an average of 1.7 seconds and instigate remedial action in a timely manner following first detection. The paper concludes with a summarisation of our main findings and discussion of potential future works which may be carried out to allow the effective detection and prevention of ransomware on systems and networks.

Keywords: IDS, IPS, IDPS, Ransomware, WannaCry, CrySiS/Dharma

1.Introduction

1.1 Introduction

In our previous paper (Wood & Eze, 2020), we examined the way in which ransomware interacts with the system on infection to implicate upon both data and system functionality. Our key finding was that it was possible to restore data and system functionality following ransomware infection. This paper iterates on our previous work (Wood & Eze, 2020) and explores the prospect of profiling the behaviour of ransomware and developing an Intrusion Detection and Prevention System (IDPS) system based exclusively on the commonly occurring system behaviours of ransomware. Section two provides an overview of ransomware in recent times, section three summarises previous research in this area, section four summarises our behavioural analysis of WannaCry and CrySiS/Dharma, section five outlines and details the proposed Talos system and its detection performance. Section six concludes the paper with a summary and discussion of this study's main findings, before drawing the paper to a close with an overview of areas requiring further work to advance the state-of-the-art in IDPS technology.

1.2 Relevant terminologies

This paper refers to several acronyms and terminologies throughout, these are Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), Intrusion Detection and Prevention Systems (IDPS) and Ransomware. Firstly, Intrusion Detection Systems (IDS) automate the process of manual intrusion detection processes by monitoring networks and systems for malicious/suspicious activity in violation of established security policies. If activity is identified, activity is logged and alerts sent to administrators (Azhagiri *et al*, 2015). Comparatively, Intrusion Prevention Systems (IPS) share the same capability of an IDS but additionally respond to and take remedial action when malicious activity is identified, before notifying administrators of the activity detected and remedial action taken (Azhagiri *et al*, 2015). An Intrusion Detection and Prevention System (IDPS) as the name implies combines the capabilities of both IDS and IPS to formulate a more robust system. Ransomware refers to a type of malicious software which is designed to restrict access to a computer system and its data until such a time a monetary fee is paid.

2. Background

As technology evolves to become more advanced and sophisticated, so have the attackers, who are continually designing and developing ever more destructive and imaginative means of breaching network and system security. As society, the economy and critical infrastructure, increasingly depend upon information technology (IT), cyberattacks are becoming increasingly attractive to attackers with potentially disastrous consequences (Jang-Jaccard & Nepal, 2014). The COVID pandemic has exacerbated the growing issue of malware/ransomware attacks, due to organisations swiftly adapting business infrastructures, which has left multiple loopholes within

IT systems, presenting attackers with easy opportunities for exploitation (Check Point, 2020). As of Q3 2020, a 50% average daily increase of attacks globally was observed, compared to the first half of 2020, specifically, the USA saw a 98.1% increase, India a 39.2% increase, Sri Lanka a 43.6% increase, Russia a 57.9% increase and Turkey a 32.5% increase (Check Point, 2020). Furthermore, 90% of security professionals report growing volumes of cyberattacks over the previous 12 months in 2020, with 4/5 reporting attacks are growing more sophisticated than 2019 (Bannister, 2020).

Ransomware remains a prevalent cybersecurity threat, capable of causing serious disruption globally. In 2017, the WannaCry ransomware affected the United Kingdom's NHS and spread rapidly across NHS networks, causing unprecedented disruption and resulting in an inability to provide patient care, with 34 trusts locked out of devices, and 46 reporting interruption (Smart, 2018). This incurred costs of £92,000,000 and resulted in 19,000 appointments being cancelled (Goud, 2018).

Petya, also caused substantial disruption in 2017. Merck Pharmaceuticals experienced disruption to research, manufacture, and product sales, amounting to damages of $670,000,000 (Davis, 2017). Whilst shipping companies, Maersk and TNT suffered substantial interference to operations and incurred nine figure costs (Greenberg, 2018). To recover, Maersk needed to reinstall 4000 servers, 45,000 systems and 2,500 applications over a 10-day recovery operation, which Maersk warns could incur losses amounting to $300 million due to severe disruption (Osborne, 2018).

Another variant, CrySiS/Dharma, has become increasingly active recently, with activity increasing 148% from February to April 2019 (Arntz, 2019). Throughout 2018, new CrySiS/Dharma variants were discovered from January to August 2018 with further increases from September to November 2018 (Coveware, 2018). If payment is made, chances of decryption range from 25% to 100%, due to some variants being more sophisticated than others (Coveware, 2018). Evidence suggests CrySiS/Dharma is becoming more sophisticated with perpetrators quashing issues which allowed decryption without payment (Nadeeau, 2018), suggesting active interest in developing increasingly destructive ransomware.

3. Previous Work

Previous studies have explored the prospect of building an IDPS for the detection and prevention of ransomware. Firstly, Azer & El-Kosiary (2018), proposed an IDPS model for detecting network intrusions and ransomware, which builds upon the concept of honeypots. Multiple decoy files are placed on the system in areas not ordinarily accessed by legitimate users, decoy files are then monitored for any access attempts. The proposed IDPS model is tested in its ability to detect a variety of ransomware types such as Cryptowall, Kovter, Winlock, Cryptolocker, Filecoder and Reveton. Samples of each ransomware are then executed on a system monitored by the IDPS model. This model detected all variants with the slowest detection time being Filecoder at 25 seconds, whilst the fastest was Cryptolocker at 15 seconds (Azer & El-Kosiary, 2018). The model is also capable of detecting attacker intrusions, using techniques such as decoy tokens, decoy servers, decoy partitions and decoy shared folders. The model upon testing, could detect all attacker intrusions, with the slowest detection time being 13 minutes whilst the fastest was 5 minutes (Azer & El-Kosiary, 2018). Evidently, whilst the system could detect all intrusions and ransomware, its detection times varied, meaning intrusions and ransomware threats are left momentarily uninterrupted, which is undesirable, hence further work is required to reduce detection times.

Celdrán *et al* (2019) developed a system intended to detect and prevent ransomware from spreading within Integrated Clinical Environments (ICE). The system utilises machine-learning (ML) techniques to detect and classify the propagation phase of ransomware attacks, whilst Network Function Visualisation (NFV) and Software Defined Network (SDN) paradigms are implemented to prevent ransomware from spreading by isolating and replacing infected network devices. Celdrán *et al* (2019) performed tests which showed the system can detect ransomware such as WannaCry, BadRabbit, Petya and PowerGhost. This is achieved by performing anomaly detection using techniques such as One-class Support Vector Machine (OC-SVM), Local Outlier Factor (LOF) and Isolation Forest (IF), whilst techniques such as; Neural Networks (NN), Naïve Bayes (NB) and Random Forest (RF) are used for classification (Celdrán *et al*, 2019). Tests are performed for each technique, with OC-SVM proving most effective for initial attack detection with 92.32% precision and 99.97% recall, whilst NB was most effective in botnet attack classification with 99.99% accuracy within 0.22 seconds (Celdrán *et al,* 2019).

4. Ransomware analysis

4.1 Ransomware sample acquisition

To allow development of Talos, common ransomware behaviours needed to be ascertained. To achieve this, several recent strains of WannaCry and CrySiS/Dharma are selected and acquired from www.virusshare.com to allow hybrid static/dynamic analysis (Table 1). Samples are populated onto a VMWare workstation virtual machine (VM) running Windows 7 SP1 for analysis.

Table 1: Acquired WannaCry and CrySiS/Dharma samples.

Family	Sample No	SHA-256 Hash
WannaCry	1	593bbcc8f34047da9960b8456094c0eaf69caaf16f1626b813484207df8bd8af
	2	ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa
	3	32f24601153be0885f11d62e0a8a2f0280a2034fc981d8184180c5d3b1b9e8cf
	4	24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c
CrySiS/Dharma	1	4b8271802c7cfec3b5258b581f4cb871edcc0c7bfb3bb7621707bdca094049a0
	2	46082f602558d2588eb9d2ab4da3efe5d5e0a7c7ef3a4812daa9b60d35fa5e63
	3	5837daaf4f7cf7280ec0a749e161015c1de39b35fa26710ce7bb22e352725ed4
	4	e264b1a0c00bcb0329845d7155bd540dfe3909f8bf72d2572db0f56bdcbb99ed
	5	f5faccb90ba57b9c4848764055d26f5ed472c84c95c48f940a6bb140b44e961b

4.2 WannaCry analysis

With reference to our previous study (Wood & Eze, 2020), it was established the first WannaCry sample when executed would create several files within its working directory (Figure 1), before creating encrypted duplicates of files prior to erasing the originals (Figure 2). It then creates several files in directories where files are encrypted, before presenting a user interface (UI) which demands a ransom and allows decryption of some files before payment (Figure 3).

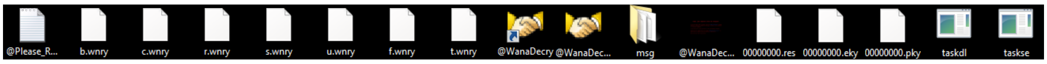
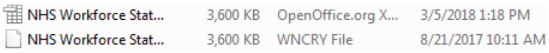
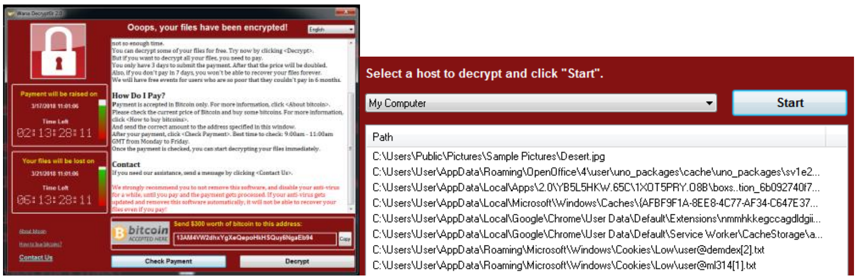


Figure 1: Created files within working directory



NHS Workforce Stat...	3,600 KB	OpenOffice.org X...	3/5/2018 1:18 PM
NHS Workforce Stat...	3,600 KB	WNCRY File	8/21/2017 10:11 AM

Figure 2: Encrypted duplicate of file, prior to originals deletion



not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?
Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am GMT from Monday to Friday.
Once the payment is checked, you can start decrypting your files immediately.

Contact
If you need our assistance, send a message by clicking <Contact Us>.

We strongly recommend you to not remove this software, and disable your anti-virus for a while, until you pay and the payment gets processed. If your anti-virus gets updated and removes this software automatically, it will not be able to recover your files even if you pay!

Payment will be raised on
3/17/2018 11:01:06
Time Left
02:13:28:11

Your files will be lost on
3/21/2018 11:01:06
Time Left
06:13:28:11

Send $300 worth of bitcoin to this address:
13AM4VW2dhxYgXeQepoHkHSQuy6NgaEb94

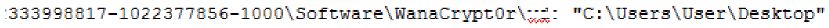
Select a host to decrypt and click "Start".

My Computer

Path
C:\Users\Public\Pictures\Sample Pictures\Desert.jpg
C:\Users\User\AppData\Roaming\OpenOffice\4\user\uno_packages\cache\uno_packages\sv1e2...
C:\Users\User\AppData\Local\Apps\2.0\YB5L5HKW.65C\1XOT5PRY.08B\boxs..tion_6b092740f7...
C:\Users\User\AppData\Local\Microsoft\Windows\Caches\{AFBF9F1A-8EE8-4C77-AF34-C647E37...
C:\Users\User\AppData\Local\Google\Chrome\User Data\Default\Extensions\nmmhkkegccagdldgii...
C:\Users\User\AppData\Local\Google\Chrome\User Data\Default\Service Worker\CacheStorage\a...
C:\Users\User\AppData\Roaming\Microsoft\Windows\Cookies\Low\user@demdex[2].txt
C:\Users\User\AppData\Roaming\Microsoft\Windows\Cookies\Low\user@ml314[1].txt

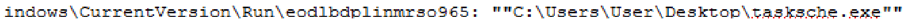
Figure 3: WannaCry user interface and sample decryption interface

Our previous study also indicated the Windows registry is modified to define a working directory for WannaCry (Figure **4**), and to set the executable 'tasksche.exe' to run automatically (Figure **5**). A wallpaper is also set and enforced, notably both files originate from the defined working directory. The referenced "tasksche.exe" executable however did not exist in the first sample, although it is observed as a process in the other samples.



```
:333998817-1022377856-1000\Software\WanaCrypt0r\[illegible]: "C:\Users\User\Desktop"
```

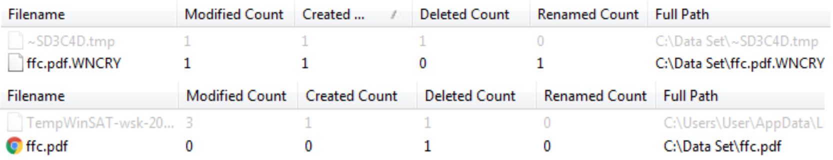
Figure 4: WannaCry defines working directory



```
indows\CurrentVersion\Run\eodlbdplinmrso965: ""C:\Users\User\Desktop\tasksche.exe""
```

Figure 5: "tasksche.exe" set to run from working directory

To assess the effects on data, the encryption behaviour of the WannaCry samples was monitored with FolderChangesView. This indicates files are not encrypted directly, but rather encrypted duplicates of files created before the originals are deleted. This behaviour is shown with the file "ffc.pdf" where an encrypted duplicate is firstly created before the original is erased (Figure 6), indicating files are recoverable following alleged encryption as established in our previous paper (Wood & Eze, 2020).

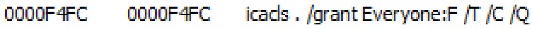


Filename	Modified Count	Created ...	Deleted Count	Renamed Count	Full Path
~SD3C4D.tmp	1	1	1	0	C:\Data Set\~SD3C4D.tmp
ffc.pdf.WNCRY	1	1	0	1	C:\Data Set\ffc.pdf.WNCRY

Filename	Modified Count	Created Count	Deleted Count	Renamed Count	Full Path
TempWinSAT-wsk-20...	3	1	1	0	C:\Users\User\AppData\L
ffc.pdf	0	0	1	0	C:\Data Set\ffc.pdf

Figure 6: Encrypted duplicate of "ffc.pdf" created and original later deleted

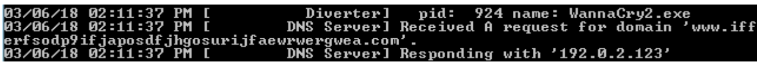
Regarding files/data, static analysis with PEiD indicated WannaCry modifies file access permissions by processing the icacls command (Figure 7). If utilised within the C:\ directory, every user would have access to all files (Plett & Poggemeyer, 2017), under which WannaCry runs as a process, posing serious implications for file integrity.



```
0000F4FC    0000F4FC    icacls . /grant Everyone:F /T /C /Q
```

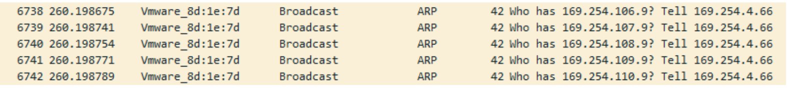
Figure 7: References to icacls

The second sample revealed notable network activity, specifically FakeNet-NG indicated the sample during execution attempts to connect to a unknown domain (Figure 8). Which, Newman (2017) argues, acts as a kill switch. Furthermore, ARP protocol traffic is observable during analysis with Wireshark (Figure 9), this behaviour is exhibited by all samples except for the first, analysis revealed such behaviour exists to find other potentially vulnerable hosts on the network to infect.



```
03/06/18 02:11:37 PM [            Diverter]   pid:  924 name: WannaCry2.exe
03/06/18 02:11:37 PM [          DNS Server] Received A request for domain 'www.iff
erfsodp9ifjaposdfjhgosurijfaewrwergwea.com'.
03/06/18 02:11:37 PM [          DNS Server] Responding with '192.0.2.123'
```

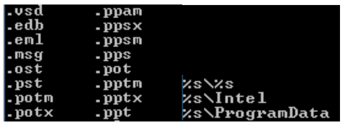
Figure 8: WannaCry requests kill switch domain



```
6738 260.198675   Vmware_8d:1e:7d   Broadcast     ARP     42 Who has 169.254.106.9? Tell 169.254.4.66
6739 260.198741   Vmware_8d:1e:7d   Broadcast     ARP     42 Who has 169.254.107.9? Tell 169.254.4.66
6740 260.198754   Vmware_8d:1e:7d   Broadcast     ARP     42 Who has 169.254.108.9? Tell 169.254.4.66
6741 260.198771   Vmware_8d:1e:7d   Broadcast     ARP     42 Who has 169.254.109.9? Tell 169.254.4.66
6742 260.198789   Vmware_8d:1e:7d   Broadcast     ARP     42 Who has 169.254.110.9? Tell 169.254.4.66
```

Figure 9: ARP activity from WannaCry observed within Wireshark

Further static analysis of the first sample was carried out with Strings. This revealed references to 3 specific directories and 177 filetypes (Figure 10). Analysis indicated; these are the filetypes that WannaCry encrypts whilst the referenced directories appear to be excluded from the encryption process.

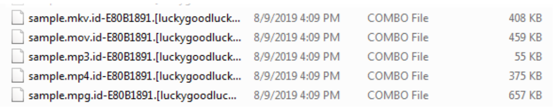


```
.vsd      .ppam
.edb      .ppsx
.eml      .ppsm
.msg      .pps
.ost      .pot
.pst      .pptm     %s\%s
.potm     .pptx     %s\Intel
.potx     .ppt      %s\ProgramData
```

Figure 10: Filetypes and directories referenced within first WannaCry sample

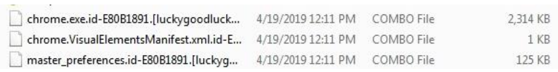
4.3 CrySiS/Dharma analysis

CrySiS/Dharma was also analysed, which upon execution will request administrator privileges, if granted, this results in immediate encryption of network drives (Figure 11), before data with the “C:\” directory and its subdirectories are encrypted (Figure 12). All variants create the “FILES ENCRYPTED.txt” file, containing a ransom/instruction note with alternating contact addresses in multiple locations. Finally, an interface is displayed demanding a ransom (Figure 13).



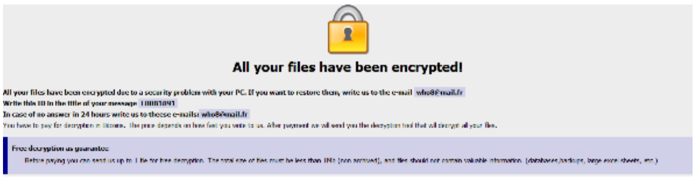
sample.mkv.id-E80B1891.[luckygoodluck...	8/9/2019 4:09 PM	COMBO File	408 KB
sample.mov.id-E80B1891.[luckygoodluck...	8/9/2019 4:09 PM	COMBO File	459 KB
sample.mp3.id-E80B1891.[luckygoodluck...	8/9/2019 4:09 PM	COMBO File	55 KB
sample.mp4.id-E80B1891.[luckygoodluck...	8/9/2019 4:09 PM	COMBO File	375 KB
sample.mpg.id-E80B1891.[luckygoodluc...	8/9/2019 4:09 PM	COMBO File	657 KB

Figure 11: Mapped network drive contents encrypted



chrome.exe.id-E80B1891.[luckygoodluck...	4/19/2019 12:11 PM	COMBO File	2,314 KB
chrome.VisualElementsManifest.xml.id-E...	4/19/2019 12:11 PM	COMBO File	1 KB
master_preferences.id-E80B1891.[luckyg...	4/19/2019 12:11 PM	COMBO File	125 KB

Figure 12: Contents of “C:\Program Files\” subdirectory encrypted



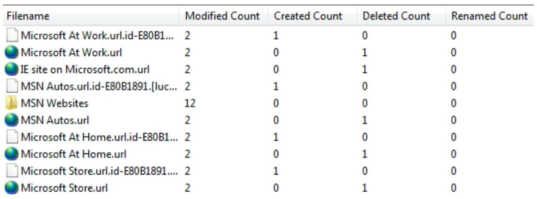
All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail who8@mail.fr
Write this ID in the title of your message E80B1891
In case of no answer in 24 hours write us to theese e-mails: who8@mail.fr
You have to pay for decryption in Bitcoins. The price depends on how fast you write to us. After payment we will send you the decryption tool that will decrypt all your files.

Free decryption as guarantee
Before paying you can send us up to 1 file for free decryption. The total size of files must be less than 1Mb (non archived), and files should not contain valuable information. (databases,backups, large excel sheets, etc.)

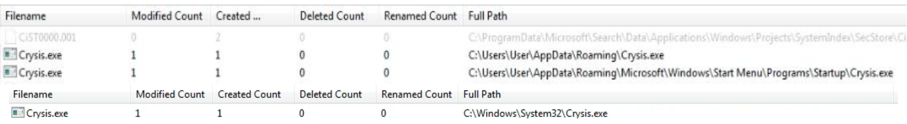
Figure 13: CrySiS/Dharma user interface

Much like WannaCry, FolderChangesView revealed, original files are copied into encrypted duplicates before the originals are modified and deleted (Figure 14). Thus, files are not encrypted directly and hence recoverable, as explored during our last paper (Wood & Eze, 2020). FolderChangesView further revealed CrySiS/Dharma will duplicate its payload into directories not normally accessed by the user on the system such as “AppData” and “System32” (Figure 15). RegShot revealed registry keys referencing such payloads to allow automatic execution (Figure 16). Thereby indicating CrySiS/Dharma is very much intent on maintaining its persistence on the system.



Filename	Modified Count	Created Count	Deleted Count	Renamed Count
Microsoft At Work.url.id-E80B1...	2	1	0	0
Microsoft At Work.url	2	0	1	0
IE site on Microsoft.com.url	2	0	1	0
MSN Autos.url.id-E80B1891.[luc...	2	1	0	0
MSN Websites	12	0	0	0
MSN Autos.url	2	0	1	0
Microsoft At Home.url.id-E80B1...	2	1	0	0
Microsoft At Home.url	2	0	1	0
Microsoft Store.url.id-E80B1891....	2	1	0	0
Microsoft Store.url	2	0	1	0

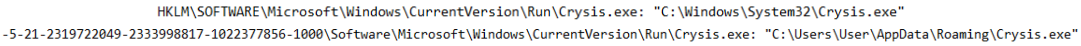
Figure 14: Original files observed to be copied into encrypted duplicates, before being modified and deleted



Filename	Modified Count	Created ...	Deleted Count	Renamed Count	Full Path
CiST0000.001	0	2	0	0	C:\ProgramData\Microsoft\Search\Data\Applications\Windows\Projects\SystemIndex\SecStore\Ci
Crysis.exe	1	1	0	0	C:\Users\User\AppData\Roaming\Crysis.exe
Crysis.exe	1	1	0	0	C:\Users\User\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Crysis.exe

Filename	Modified Count	Created Count	Deleted Count	Renamed Count	Full Path
Crysis.exe	1	1	0	0	C:\Windows\System32\Crysis.exe

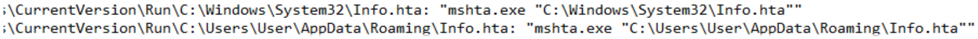
Figure 15: Payload duplicated into discreet directories



```
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\Crysis.exe: "C:\Windows\System32\Crysis.exe"
-5-21-2319722049-2333998817-1022377856-1000\Software\Microsoft\Windows\CurrentVersion\Run\Crysis.exe: "C:\Users\User\AppData\Roaming\Crysis.exe"
```

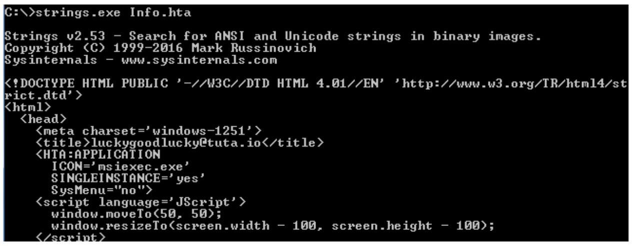
Figure 16: Registry keys created to Run duplicated payloads

Notably, “Info.hta” is created across all samples within the same directory as the duplicated payload and the registry modified to automatically execute this file with “mshta.exe” as indicated by RegShot (Figure 17). Analysis of “Info.hta” with the Strings tool revealed it to contain a series of HTML and JavaScript code (Figure 18). Further static analysis of the samples with Immunity Debugger revealled references to WebDav in the form of “davcint” (Figure **19**), a tool to establish and manage secure connections to remote WebDav servers (Microsoft, 2019). This may suggest CrySiS/Dharma can spread to or access data on remote servers on the network/internet.



```
;\CurrentVersion\Run\C:\Windows\System32\Info.hta: "mshta.exe "C:\Windows\System32\Info.hta""
;\CurrentVersion\Run\C:\Users\User\AppData\Roaming\Info.hta: "mshta.exe "C:\Users\User\AppData\Roaming\Info.hta""
```

Figure 17: Registry key to automatically execute “Info.hta” with “mshta.exe”

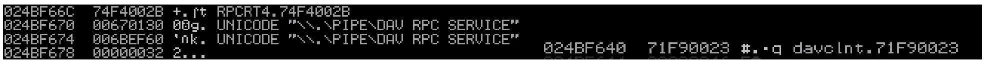


```
C:\>strings.exe Info.hta

Strings v2.53 - Search for ANSI and Unicode strings in binary images.
Copyright (C) 1999-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

<!DOCTYPE HTML PUBLIC '-//W3C//DTD HTML 4.01//EN' 'http://www.w3.org/TR/html4/st
rict.dtd'>
<html>
  <head>
    <meta charset='windows-1251'>
    <title>luckygoodlucky@tuta.io</title>
    <HTA:APPLICATION
      ICON='msiexec.exe'
      SINGLEINSTANCE='yes'
      SysMenu="no">
    <script language='JScript'>
      window.moveTo(50, 50);
      window.resizeTo(screen.width - 100, screen.height - 100);
    </script>
```

Figure 18: Contents of “Info.hta” as revealed with the Strings tool



```
024BF66C  74F4002B +.ṛt RPCRT4.74F4002B
024BF670  00670130 00g. UNICODE "\\.\PIPE\DAV RPC SERVICE"
024BF674  006BEF60 'nk. UNICODE "\\.\PIPE\DAV RPC SERVICE"
024BF678  00000032 2...                                      024BF640  71F90023 #.·q davclnt.71F90023
```

Figure 19: References to WebDav observed within CrySiS/Dharma

5.Proposed Talos IDPS system

5.1 Introduction

In this section of the paper, we present the prototype Talos IDPS system and discuss some of the ransomware behavioural features selected from the earlier ransomware analysis and components which have been developed to construct Talos. The naming of the Talos prototype system, is taken from Greek mythology and is named after the giant automaton who defended Europa in Crete from pirates and invaders by circling the island three times daily and hurling boulders at approaching enemy ships.

5.2 Common features of ransomware

After ascertaining the common behaviours of the ransomware samples, it was evident, many indicators of a ransomware infection manifested within the windows filesystem and as system processes. Thus, as a starting point for Talos, a selection of key filesystem features for both ransomware families are selected (Table 2). Notably, other behaviours, specifically file integrity issues may be generalisable to other ransomware variants, this aspect however will be explored as part of our future work.

Table 2: WannaCry and CrySiS/Dharma sample common features

WannaCry	**File Types**	.wnry		.wncryt	.wncry	
	Processes	tasksche.exe		taskse.exe	taskdl.exe	
	Created Files	C:\Users\User\Desktop\@WanaDecryptor@.bmp			C:\@Please_Read_Me@.txt	
		C:\Users\User\Desktop\@WanaDecryptor@.exe			C:\@WanaDecryptor@.exe	
		C:\Users\User\Desktop\@Please_Read_Me@.txt			C:\Windows\tasksche.exe	
CrySiS/Dharma	**File Types**	.combo	.HARMA	.PLEX	.2020	.aa1
	Processes	mshta.exe				
	Created Files	C:\Users\User\Desktop\FILES ENCRYPTED.txt			C:\Windows\System32\Info.hta	
		C:\Users\User\AppData\Roaming\Info.hta			C:\FILES ENCRYPTED.txt	

5.3 Talos Prototype components

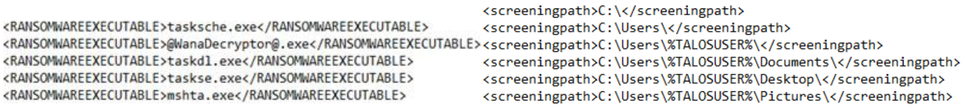
Considering information ascertained during analysis, and to allow development of Talos, it was determined several key components would be required, which included a main component, a file integrity check, a filetype check, a blacklisted file check, process check and reset/initialisation components. Each component was created in Python, the functions of each are explained in Table 3.

Table 3: Descriptions of each component of Talos

Component	Description
Main	The main component sequentially calls and executes each component and interprets the output of each i.e., status codes and takes remedial action where required. This is achieved by a function known as the “watcher” and the use of a centralised concern level variable, which is raised in the event activity is detected. If the variable raises to 1, the function disables all network adapters to prevent ransomware propagation, whereas if raising to 2 or above, the main component ensures all network adapters are disabled and the system is safely shut down to eliminate the risk of further damage. The main component will also call the initialisation and reset components if honey file integrity is reported as damaged by the “File Integrity Check” component.
Initialisation	The initialisation component, when called, generates multiple “.txt” honey files across various system directories, where ransomware activity is known to occur, such as the; “C:\”, user’s directory amongst other strategic locations. Such files contain a series of ASCII characters and digits of randomised lengths, to evade detection by malicious software. Upon generation, SHA256 and the paths/locations for each file are saved, to allow later recall and integrity checks by the File Integrity Check component.
File Integrity Check	Ransomware commonly affects file integrity, either through deletion or modification. This component verifies file integrity by gathering a list of files generated earlier by the initialisation component and their SHA256 hashes, it then generates new SHA256 hash for each of the files and compares them against the earlier generated SHA256 hashes. If these hashes match, then no file integrity issues have occurred, whereas if it does not match or files no longer exist, this strongly indicates file integrity is impacted, in this instance the component will raise a status code for remediation by the main component.
Filetype Check	This component examines defined directories for filetypes associated with ransomware, to achieve this, a list of files within each directory is gathered, and checks performed to verify if any end with a defined ransomware extension. If found, file details are logged and a status code raised, to allow later remediation.
Process Check	This component gathers a list of running system processes and checks this list to establish if it contains the name of an associated ransomware process loaded from a file. If a process is found to be running, the component will generate a log and raise an alarm for interpretation by the main component.
Blacklisted File Check	During execution, ransomware creates multiple files in various locations across the system, containing payloads or other required data. This component examines various defined directories for the presence of such files. If files are found, the component raises an alarm for remediation by the main component.
Reset	If the file integrity check reports that a generated honey file is damaged, then the reset component is called by the main component to erase all previously generated files before the initialisation component is called to generate a new selection of honey files for further integrity checks to be performed.

5.4 Data

One of Talos’s key features is the consideration of known ransomware behaviours to detect activity, to permit this, data on ransomware behaviour is stored within files. This data includes the list of files/hashes generated by the initialisation component, the processes, files and filetypes associated with ransomware and the resulting status codes of each Talos component amongst other data. Each data item is stored within XML tags (Figure 20), which is later retrieved through regular expression parsing by each component. This method of storing and retrieving data, and the modularity of Talos will permit later adaptation of the system to allow the detection of further malware/ransomware variants.



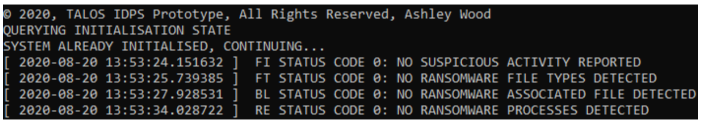
```
<RANSOMWAREEXECUTABLE>tasksche.exe</RANSOMWAREEXECUTABLE>
<RANSOMWAREEXECUTABLE>@WanaDecryptor@.exe</RANSOMWAREEXECUTABLE>
<RANSOMWAREEXECUTABLE>taskdl.exe</RANSOMWAREEXECUTABLE>
<RANSOMWAREEXECUTABLE>taskse.exe</RANSOMWAREEXECUTABLE>
<RANSOMWAREEXECUTABLE>mshta.exe</RANSOMWAREEXECUTABLE>
```

```
<screeningpath>C:\</screeningpath>
<screeningpath>C:\Users\</screeningpath>
<screeningpath>C:\Users\%TALOSUSER%\</screeningpath>
<screeningpath>C:\Users\%TALOSUSER%\Documents\</screeningpath>
<screeningpath>C:\Users\%TALOSUSER%\Desktop\</screeningpath>
<screeningpath>C:\Users\%TALOSUSER%\Pictures\</screeningpath>
```

Figure 20: Two example of files from where components will read data from

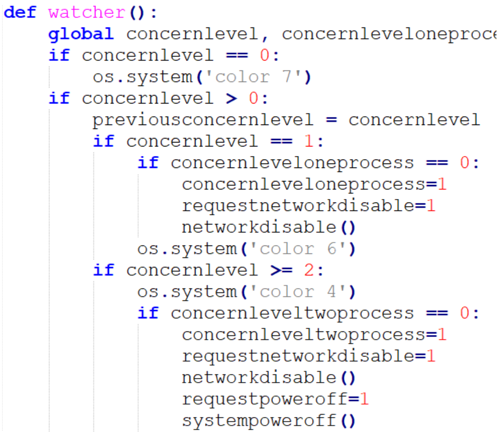
5.5 Prototype system

Upon compiling data on common ransomware behaviours and establishing the required detection mechanisms, all required components of Talos could be developed. Once the system was built; it could then be executed as a console application (Figure 21). Talos executes each component sequentially as part of the main loop and interprets the results at each stage for action to be taken where necessary. If a component detects a single event, then a centralised “concernlevel” variable is incremented by “1”, whereas if a component reports multiple events, it is set to “2”. A function known as the “watcher” will check the value of this variable after executing each component, if it reaches “1”, all network adapters are disabled to minimise the spread of ransomware, whereas if greater than or equal to “2”, all network adapters are disabled and the system safely powered off to prevent further damage, as shown within the code snippet (Figure 22).



```
© 2020, TALOS IDPS Prototype, All Rights Reserved, Ashley Wood
QUERYING INITIALISATION STATE
SYSTEM ALREADY INITIALISED, CONTINUING...
[ 2020-08-20 13:53:24.151632 ]  FI STATUS CODE 0: NO SUSPICIOUS ACTIVITY REPORTED
[ 2020-08-20 13:53:25.739385 ]  FT STATUS CODE 0: NO RANSOMWARE FILE TYPES DETECTED
[ 2020-08-20 13:53:27.928531 ]  BL STATUS CODE 0: NO RANSOMWARE ASSOCIATED FILE DETECTED
[ 2020-08-20 13:53:34.028722 ]  RE STATUS CODE 0: NO RANSOMWARE PROCESSES DETECTED
```

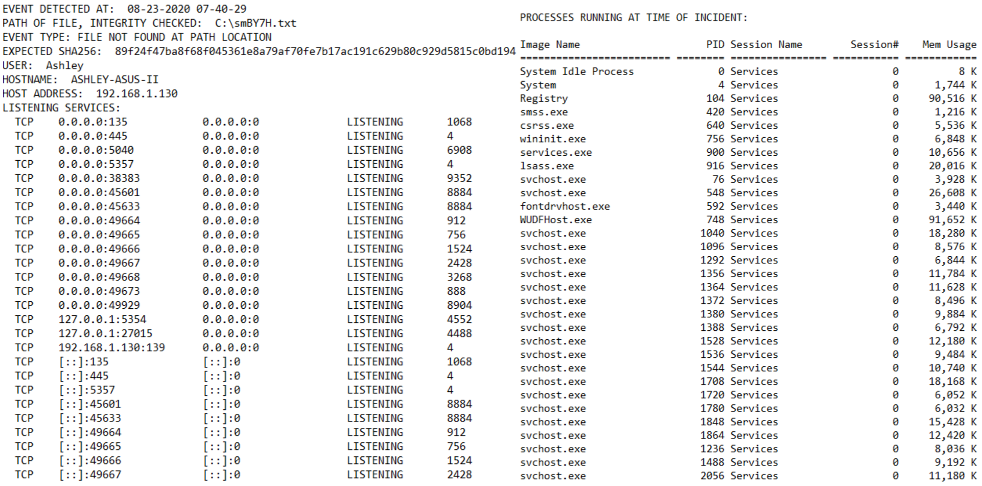
Figure 21: Main component successfully running all components



```
def watcher():
    global concernlevel, concernleveloneproc
    if concernlevel == 0:
        os.system('color 7')
    if concernlevel > 0:
        previousconcernlevel = concernlevel
        if concernlevel == 1:
            if concernleveloneprocess == 0:
                concernleveloneprocess=1
                requestnetworkdisable=1
                networkdisable()
            os.system('color 6')
        if concernlevel >= 2:
            os.system('color 4')
            if concernleveltwoprocess == 0:
                concernleveltwoprocess=1
                requestnetworkdisable=1
                networkdisable()
                requestpoweroff=1
                systempoweroff()
```

Figure 22: The “watcher” function (source code)

In addition to performing remedial action, each component of Talos will create respective logs of the detected activity, for example the File Integrity check component, will record details of the activity detected, the expected and generated SHA256 hashes and additionally details of the current user, IP address, running processes and listening network services (Figure 23). All of which are collected for examination at a later point in time to ascertain precisely what occurred on the system to cause the activity.



```
EVENT DETECTED AT:  08-23-2020 07-40-29
PATH OF FILE, INTEGRITY CHECKED:  C:\smBY7H.txt
EVENT TYPE: FILE NOT FOUND AT PATH LOCATION
EXPECTED SHA256:  89f24f47ba8f68f045361e8a79af70fe7b17ac191c629b80c929d5815c0bd194
USER:  Ashley
HOSTNAME:  ASHLEY-ASUS-II
HOST ADDRESS:  192.168.1.130
LISTENING SERVICES:
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       1068
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING       6908
  TCP    0.0.0.0:5357           0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:38383          0.0.0.0:0              LISTENING       9352
  TCP    0.0.0.0:45601          0.0.0.0:0              LISTENING       8884
  TCP    0.0.0.0:45633          0.0.0.0:0              LISTENING       8884
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING       912
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING       756
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING       1524
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING       2428
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING       3268
  TCP    0.0.0.0:49673          0.0.0.0:0              LISTENING       888
  TCP    0.0.0.0:49929          0.0.0.0:0              LISTENING       8904
  TCP    127.0.0.1:5354         0.0.0.0:0              LISTENING       4552
  TCP    127.0.0.1:27015        0.0.0.0:0              LISTENING       4488
  TCP    192.168.1.130:139      0.0.0.0:0              LISTENING       4
  TCP    [::]:135               [::]:0                 LISTENING       1068
  TCP    [::]:445               [::]:0                 LISTENING       4
  TCP    [::]:5357              [::]:0                 LISTENING       4
  TCP    [::]:45601             [::]:0                 LISTENING       8884
  TCP    [::]:45633             [::]:0                 LISTENING       8884
  TCP    [::]:49664             [::]:0                 LISTENING       912
  TCP    [::]:49665             [::]:0                 LISTENING       756
  TCP    [::]:49666             [::]:0                 LISTENING       1524
  TCP    [::]:49667             [::]:0                 LISTENING       2428
```

```
PROCESSES RUNNING AT TIME OF INCIDENT:

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
System Idle Process              0 Services                   0          8 K
System                           4 Services                   0      1,744 K
Registry                       104 Services                   0     90,516 K
smss.exe                       420 Services                   0      1,216 K
csrss.exe                      640 Services                   0      5,536 K
wininit.exe                    756 Services                   0      6,848 K
services.exe                   900 Services                   0     10,656 K
lsass.exe                      916 Services                   0     20,016 K
svchost.exe                     76 Services                   0      3,928 K
svchost.exe                    548 Services                   0     26,608 K
fontdrvhost.exe                592 Services                   0      3,440 K
WUDFHost.exe                   748 Services                   0     91,652 K
svchost.exe                   1040 Services                   0     18,280 K
svchost.exe                   1096 Services                   0      8,576 K
svchost.exe                   1292 Services                   0      6,844 K
svchost.exe                   1356 Services                   0     11,784 K
svchost.exe                   1364 Services                   0     11,628 K
svchost.exe                   1372 Services                   0      8,496 K
svchost.exe                   1380 Services                   0      9,884 K
svchost.exe                   1388 Services                   0      6,792 K
svchost.exe                   1528 Services                   0     12,180 K
svchost.exe                   1536 Services                   0      9,484 K
svchost.exe                   1544 Services                   0     10,740 K
svchost.exe                   1708 Services                   0     18,168 K
svchost.exe                   1720 Services                   0      6,052 K
svchost.exe                   1780 Services                   0      6,032 K
svchost.exe                   1848 Services                   0     15,428 K
svchost.exe                   1864 Services                   0     12,420 K
svchost.exe                   1236 Services                   0      8,036 K
svchost.exe                   1488 Services                   0      9,192 K
svchost.exe                   2056 Services                   0     11,180 K
```

Figure 23: Example of log file generated by the File Integrity Check component

5.6 Talos Prototype performance benchmarking

After building the prototype and verifying its functionality, benchmarking was performed to measure the systems detection and response times to threats, specifically the earlier analysed WannaCry and CrySiS/Dharma samples. To perform benchmarking, a testing script was prepared, which; executes each ransomware sample, launches Talos and records the execution times of each. Benchmarking results indicated Talos could detect all variants of WannaCry and CrySiS/Dharma promptly (Table 4), with an average first detection time of 2 seconds for WannaCry and 1.6 seconds for CrySiS/Dharma, resulting in an average first detection time of 1.7 seconds. Results also indicate Talos can initiate remedial action within a reasonable timeframe with CrySiS/Dharma, although there is evidently a need to reduce these times with WannaCry.

Table 4: WannaCry and CrySiS/Dharma sample benchmarking results

WannaCry benchmarking results	Sample	Execution Time	First Detection	NWAD	SPO
	1	11:54:37	11:54:38	11:54:47	11:54:48
	2	12:03:01	12:03:04	12:03:12	12:03:20
	3	11:43:59	11:44:01	11:44:09	11:44:10
	4	11:31:17	11:31:19	11:31:27	11:31:29
CrySiS/Dharma benchmarking results	1	12:17:09	12:17:11	12:17:13	12:17:13
	2	12:27:01	12:27:03	12:27:05	12:27:05
	3	12:34:10	12:34:12	12:34:14	12:34:14
	4	12:40:03	12:40:04	12:40:07	12:40:07
	5	12:56:23	12:56:24	12:56:25	12:56:29

Notably across all CrySiS/Dharma variants with the exception of sample 5, the network adaptor disable (NWAD) and system power off (SPO) trigger times are identical, this occurred due to the file integrity check component detecting multiple incidents i.e. a file being modified and another deleted. Which immediately sets the components status code to a higher level, resulting in the main component setting the “concernlevel” variable to “2”, which results in the “watcher” function calling both the NWAD and SPO functions.

Benchmarking further revealed performance disparity between individual components, namely, the file integrity component proved most effective at detecting CrySiS/Dharma, whilst the blacklisted file check proved most effective at detecting WannaCry, made evident by the first detection order (

Table 5). Where no result is recorded, Talos initiated remedial action before components detected activity. The ransomware associated filetype check proved least effective and only detected 1 WannaCry variant during testing. The performance disparity between each component is notable, as this indicates individual component performance is intrinsically linked to individual ransomware/malware behaviour, which suggests individual components may prove more effective at detecting one variant over another. This finding may have potential ramifications if Talos is later adapted to account for other variants, and further suggests, combining multiple components may be required.

Table 5: Individual component detection when tested against WannaCry and CrySiS/Dharma variants

	Sample	File Integrity Check	Ransomware associated filetype check	Blacklisted File check	Ransomware process check
WannaCry component detection results	1	3		1	2
	2		1	2	3
	3	3		1	2
	4	3		1	2
CrySiS/Dharma component detection results	1	1			
	2	1			
	3	1			
	4	1			
	5	1		2	

6.Conclusions and future work

6.1 Study summary

In this study, the behaviour of multiple ransomware variants was profiled using static/dynamic analysis and later analysed to develop detection mechanisms for Talos, specifically focusing on filesystem activity. The system developed in this study, has shown an IDPS utilising the common behavioural features of ransomware/malware can prove highly beneficial in the active detection and mitigation of ransomware. The Talos system could detect all ransomware variants tested promptly, averaging 1.7 seconds for first detection.

Results achieved during performance benchmarking of Talos are promising, and represent an improvement over other comparable works, such as Azer & El-Kosairy's (2018) study, where the detection time ranged from 15 seconds for Cryptolocker to 25 seconds for filecoder. Notably, Talos and the work of Azer & El-Kosairy (2018) are designed to detect different malware/ransomware types with the work of Azer & El-Kosairy's (2018) able to detect other attack and intrusion types. Furthermore, Talos falls behind systems incorporating artificial intelligence-based techniques, such as the work of Celdrán *et al* (2019) where the classification time was quicker at 0.22 seconds. Consequently, further performance benchmarking and iteration of Talos is required to fully assess its performance against comparable systems. However, Talos is modular and may be later adapted to account for further malware/ransomware variants and other attack/intrusion types.

6.2 Future Work

Whilst Talos offers a promising level of performance, there are areas which require further development to improve the accuracy and resiliency of the system. Firstly, Talos is at present designed and tested to detect several strains of CrySiS/Dharma and WannaCry, which constrains the system's ability to detect other ransomware/malware variants and other intrusive/malicious activity. Thus, more complex network propagating ransomwares such as Petya (Wood & Eze, 2020) and other forms of attacks/intrusions are not yet detected by Talos. Furthermore, the current approach taken with Talos, assumes that other security mechanisms such as firewalls and antivirus products have failed to contain threats to the point where ransomware/malware can attain a foothold on the system. To address these area, Talos will be further developed in our future work to incorporate the common behavioural characteristics of other intrusive activities and ransomware/malware variants and also to consider activity occurring on the wider network and filesystem as part of a hybrid host-based and network-based system. This will help to allow the earlier detection of threats and will drastically improve the detection capability of Talos, allowing it to detect a diverse range of attacks.

Whilst this study has specifically focused on CrySiS/Dharma and WannaCry behaviours, it is believed the behaviours uncovered may be generalised to other forms of ransomware i.e., the way in which ransomware creates encrypted duplicates of files before affecting the integrity of the originals. Other features, such as file creation, filetypes and process spawning may be generalisable to other variants, the system will however require further adaptation and testing further to account for this, this aspect will be addressed in our future work.

Another area requiring further develop is Talos's decision-making capabilities, present Talos performs two remedial actions in sequence, firstly disabling network adapters if one event is detected and secondly powering off the system if two or more events are detected. The aim of this is to prevent ransomware spreading to other systems and to prevent further damage to the system and data. This approach however is potentially problematic in the event of false-positive errors, where legitimate activity is erroneously perceived as malicious. This area is acknowledged as a problem, and will be addressed in our future work, to achieve this Artificial Intelligence (AI) will be implemented into Talos, to allow it make more informed decisions about the actions its takes by considering the characteristics of previously detected threats to determine how best to respond. AI and Machine-learning based techniques have received considerable interest in the wider-research community, Celdrán *et al* (2019) applied machine learning techniques to their proposed system and saw promising detection accuracy scores and classification times. William (2020) argues, AI-based IDS unlike traditional IDS, have capability to learn over time from previous attacks to allow creation of new detection algorithms, allowing it to learn how to stop stealthy adversaries. Our future work will aim to address these areas, to allow Talos to perform more effectively.

Whilst the performance of Talos is evidently promising, based upon the results of our own performance tests. It is acknowledged that Talos is a work in progress the rates of false-positives and false-negatives is not yet ascertained. At the present point of development, it is acknowledged that Talos could potentially perceive legitimate/benign activity as malicious and actively block it, or could fail to detect other forms of ransomware, outside of those analysed as part of this study, resulting in false-positive and false-negative errors. Our future work will measure these rates by simulating multiple benign and malicious activities on the system and recording the response of Talos to ascertain the true false-positive and false-negative error rates.

References

Arntz, P. (2019). *Threat spotlight: CrySIS, aka Dharma ransomware, causing a crisis for businesses*. Retrieved from https://blog.malwarebytes.com/threat-analysis/2019/05/threat-spotlight-crysis-aka-dharma-ransomware-causing-a-crisis-for-businesses/

Azer, M, A., & El-Kosairy, A. (2018). Intrusion and Ransomware Detection System. 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), 1-7. https://doi.org/10.1109/CAIS.2018.8471688

Azhagiri, M., Karthik, S., & Rajesh, S. (2015). INTRUSION DETECTION AND PREVENTION SYSTEM: TECHNOLOGIES AND CHALLENGES. *International Journal of Applied Engineering Research, 10*(87), 1-12. https://www.researchgate.net/publication/287208734_Intrusion_Detection_and_Prevention_System_Tchnologies_and_Challenges

Bannister, A. (2020). *Remote working during coronavirus pandemic leads to rise in cyber-attacks, say security professionals.* Retrieved from https://portswigger.net/daily-swig/remote-working-during-coronavirus-pandemic-leads-to-rise-in-cyber-attacks-say-security-professionals

Celdrán, A, H., Clemente, F, J, G., Gómez, A, L, P., Lee, I., & Weimer, J. (2019). Intelligent and Dynamic Ransomware Spread Detection and Mitigation in Integrated Clinical Environments. *Sensors 2019, 19*(5), 1-31. https://doi.org/10.3390/s19051114

Check Point. (2020). *Global Surges in Ransomware Attacks.* Retrieved from https://blog.checkpoint.com/2020/10/06/study-global-rise-in-ransomware-attacks/

Coveware. (2018). *Why New Dharma Ransomware is More Dangerous than ever.* Retrieved from https://www.coveware.com/blog/2018/11/26/why-new-dharma-ransomware-is-more-dangerous-than-ever

Davis, J. (2017). *Petya attacks now appear to be causing permanent damage*. Retrieved from https://www.healthcareitnews.com/news/petya-attacks-now-appear-be-causing-permanent-damage

Goud, N. (2018). *NHS lost £92 million and Cancelled 19K appointments due to WannaCry Ransomware Attack*. Retrieved from https://www.cybersecurity-insiders.com/nhs-lost-92-million-and-cancelled-19k-appointments-due-to-wannacry-ransomware-attack/

Greenberg, A. (2017). *THE WANNACRY RANSOMWARE HACKERS MADE SOME REAL AMATEUR MISTAKES.* Retrieved from https://www.wired.com/2017/05/wannacry-ransomware-hackers-made-real-amateur-mistakes/

Jang-Jaccard, J., & Nepal, S. (2014). A survey of emerging threats in cybersecurity. *Journal of Computer and System Sciences, 80*(5), 973-993. https://doi.org/10.1016/j.jcss.2014.02.005

Microsoft. (2019). *davclnt.h header.* Retrieved from https://docs.microsoft.com/en-us/windows/win32/api/davclnt/

Nadeau, M. (2018). *11 Ways Ransomware is Evolving.* Retrieved from https://insights.samsung.com/2018/04/11/11-ways-ransomware-is-evolving/

Newman, L, H. (2017). *HOW AN ACCIDENTAL 'KILL SWITCH' SLOWED FRIDAY'S MASSIVE RANSOMWARE ATTACK.* Retrieved from https://www.wired.com/2017/05/accidental-kill-switch-slowed-fridays-massive-ransomware-attack/

Osborne, C. (2018). *NonPetya ransomware forced Maersk to reinstall 4000 servers, 45000 PCs*. Retrieved from https://www.zdnet.com/article/maersk-forced-to-reinstall-4000-servers-45000-pcs-due-to-notpetya-attack/

Plett, C., & Poggemeyer, L. (2017). *Icacls.* Retrieved from https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/icacls

Smart, W. (2018). *Lessons learned review of the WannaCry Ransomware Cyber Attack.* Retrieved from https://www.england.nhs.uk/wp-content/uploads/2018/02/lessons-learned-review-wannacry-ransomware-cyber-attack-cio-review.pdf

William, D. (2020). *How AI can help improve intrusion detection systems.* Retrieved from https://gcn.com/articles/2020/04/15/ai-intrusion-detection.aspx

Wood, A. & Eze, T. (2020). The Evolution of Ransomware Variants. *Proceedings of the 19th European Conference on Cyber Warfare and Security ECCWS 2020* (pp. 410-420). Chester, United Kingdom: ACPI.