Threat Detection and Incident Response For Windows OS

Submitted By

Siddharth Bhatt

16MCEI01

तमसो मा ज्योतिर्गमय

NIRMA
UNIVERSITY

INSTITUTE OF TECHNOLOGY

NAAC ACCREDITED ‘A’ GRADE

DEPARTMENT OF COMPUTER ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2018

Threat Detection and Incident Response For Windows OS

Major Project

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering (Information and Network Security)

Submitted By

Siddharth Bhatt

(16MCEI01)

Guided By

Dr. Sharada Valiveti

तमसो मा ज्योतिर्गमय

NIRMA UNIVERSITY

INSTITUTE OF TECHNOLOGY

NAAC ACCREDITED ‘A’ GRADE

DEPARTMENT OF COMPUTER ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2018

Certificate

This is to certify that the major project entitled **”Threat Detection and Incident Response For Windows OS”** submitted by **Siddharth Bhatt(16MCEI01)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering (Information and Network Security) of Nirma University, Ahmedabad, is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project part-II, to the best of my knowledge, haven’t been submitted to any other university or institution for award of any degree or diploma.

Dr. Sharada Valiveti
Guide & Associate Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. Sharada Valiveti
Associate Professor,
Coordinator M.Tech - CSE (INS)
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sanjay Garg
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr Alka Mahajan
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Siddharth Bhatt**, **16MCEI01**, give undertaking that the Major Project entitled **”Threat Detection and Incident Response For Windows OS”** submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science Engineering (Information and Network Security)** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made.It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

____

Signature of Student
Date:
Place:

Endorsed by
Guide Name
(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Dr. Sharada Valiveti**, Associate Professor, Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout this work. The appreciation and continual support he has imparted has been a great motivation to me in reaching a higher goal. His guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

It gives me an immense pleasure to thank **Dr. Sanjay Garg**, Hon'ble Head of Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr. Alka Mahajan**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation she has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

- Siddharth Bhatt
16MCEI01

Abstract

Here analysis of Ransomware attacks is performed, data obtained from the analysis is be used to detect an ransomware attack in digital era for IoT. With the rise in Digital India and many more start-ups computer world had witnessed major Ransomware outbreak in May 2017 which infected more than 4,00,000 systems at time only by the malware WannaCry. This advanced ransomware has the capability to encrypts user important data, and post attack user wont be possible to recover without paying ransom amount. Generally they would ask an high ransom as demand mostly in bit-coins to unlock the device in or they would threaten to delete or may not give key to decrpty and even may increase the ransom amount to be paid. Nowadays cell phone has become immense part of humans life. The focuses is to go in depth how this malware attacks the target system and thus proves how harmfull attacker could be and also attackers demands large amount of ransom. Better approach is discussed on how to prevent this ransomware attack and some precaution for all. Data obtained from analysis also ensures the awareness of Ransomware attack, during the course of time from its origination, geographical attacking analysis and operating system based attacks mainly for Windows OS. The analysis of such malware helps us for the awareness and counter measures. Thus it will play a key role in safe use of Digital India, E-Governance, E-Commerce, IoT and so on.

Contents

Certificate **iii**

Statement of Originality **iv**

Acknowledgements **v**

Abstract **vi**

List of Figures **viii**

List of Tables **ix**

1 Introduction **1**
1.1 Threat Detection System 2
1.2 Problem Statement 3

2 Literature Survey **4**

3 Working of Ransomware **9**

4 Proposed System **12**

5 Implementation And Results **16**
5.1 Tools and Technology 16
5.2 System Configuration 16
5.3 R-Locker 17

6 Conclusion and Future Work **22**

Bibliography **23**

List of Figures

1.1 Malware Statistics 2

3.1 Ransomware CC Server Connection 11

4.1 Flow Diagram of The System 14
4.2 R-Locker Working[1] 15

5.1 R Locker Working [1] 17
5.2 Named Pipe Server Creation 18
5.3 Named Pipe Server 19
5.4 Named Pipe Client 19
5.5 Directory Watcher 20
5.6 Incremental Safe Backup 21

List of Tables

2.1 Survey of Research Papers 8

Chapter 1

Introduction

Ransomware is a kind of Malware that taints the individual documents of the client and does not allow to access until payment is paid. The asked for payoff installment is normally in the request of a couple of hundreds US dollars (or equal in crypto or generally untraceable money). Unmistakably, the accomplishment of these assaults relies on upon whether the greater part of the casualties consents to pay (e.g., on account of the dread of losing their information). From a specialized perspective, Ransomware families are currently much progressed. While original Ransomware were cryptographically feeble, the Current families scramble each document with a one of a kind symmetric key ensured by Open key cryptography. Subsequently, the odds of an actively recuperation (without Paying the payoff) have radically diminished. More than 4,000 Ransomware assaults have happened each day since the start of 2016. That is a 300% expansion more than 2015, where 1,000 Ransomware assaults were seen every day. 56,000 Ransomware diseases in March 2016. This is type of malicious software and different kind of ransomware are found till date.

- Lock screen ransomware (WinLocker Ransomware)
- Crypto ransomware (File Encryptor Ransomware)

From a specialized perspective, ransomware families are presently very progressed. While original ransomware were cryptographically feeble, the current families scramble each record with a one of a special symmetric key secured by public-key cryptography. Subsequently, the chance for recovering (without paying the payment) have radically decreased, World Wide malware statistics from Symantec Lab is provided in.[?]

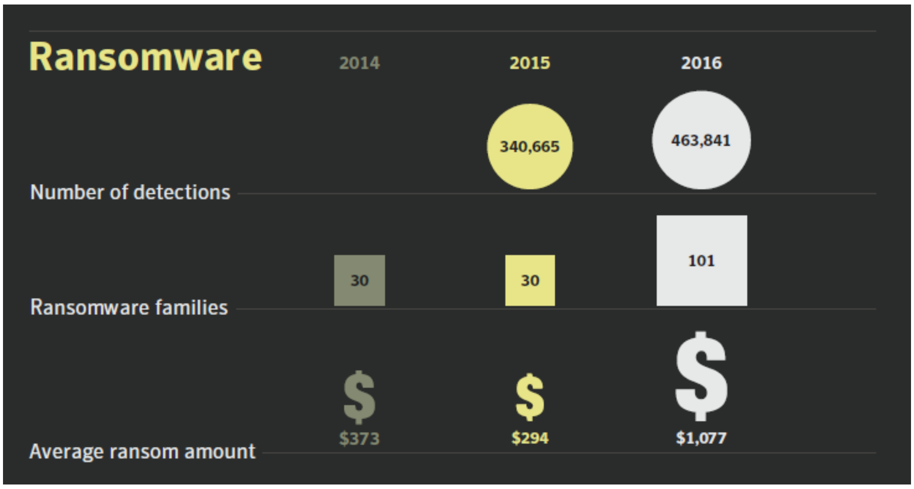


Figure 1.1: Malware Statistics

In this paper, we will take a look at where and when the Ransomware attacks worked, not just from a geographical point of view but also from operating system viewpoint. We will also look at how these threats evolved, what factors are at play to make Ransomware the major problem that it is today, and where Ransomware is likely to surface next. Ransomware outbreak happened in May 2017 affecting more than 4,00,000 machine only with its one attack mechanism malware called as Wanna Cry while Petya Ransomware was also hit the market after some days affecting many user causing a situation where user could not probably recover back the data.

1.1 Threat Detection System

It has been an challenge to detect the ransomware attacks. Attackers smartly target the Vulnerability of the existing software and server and thus user gets into trap and becomes victim thus at place of either losing data or pay ransom amount of around $300 to 1000$ are per data in the system. Threat detection as techniques to detect the attack either by the scanning payload or by the means of detecting the nature of attack from the statistics of the network. Ransomware attackers are so smart they had targeted Eternal Blue exploit to target the system thus user has no option to prevent it self from infection. They had targeted SMBv2 exploit of Microsoft windows to inject shell code into system directly with the help of an IP address of user with dynamic code from server thus making it more difficult for the system admin to detect the attack. They targeted attack through backdoor double pulsar that is undetected since a long time and thus it was used to inject into vulnerable system and even spread it to network connected with that device.

1.2 Problem Statement

With the advancements in the malware technology we need a threat Detection and incident response system, that help us to detect and prevent Zero Day attacks. We need light weight system which does not require training nor is heavy to ram. This system should be able to implement layered defense approach. This would help us to defend outbreak such as WannaCry or Petya.

Chapter 2

Literature Survey

This section covers the examination and business related to this theme. A bunches of continuous research are completed for the threat detection and incident response. Most of the Zero day detection system focuses on threat detection will Machine learning and artificial intelligence.

While machine learning needs a lot of training the model or training the set. While we need light weight approach, which is light, effective and low in consumption.

Our approach is to focus on R-locker: thwarting ransomware action through a honeyfile-based approach Gomez-Hernandez, JA and Alvarez-Gonzalez, L and Garca-Teodoro Elsevier This paper presents a novel approach intended not just to early detect ransomware but to completly thwart its action. For that,a set of honeyfiles are deployed around the target environment in order to catch the ransomware. In addition to frustrate its action, our honeyfile solution is able to automatically launch countermeasures to solve the infection. Moreover, as it does not require previous training or knowledge, the approach allows fighting against unknown, zero-day ransomware related attacks. As a proof of concept, we have developed the approach for Linux platforms called as R-Locker.

Paper	Author	Publication	Important Points
UNVEIL: A Large-Scale, Automated Approach to Detecting Ransomware	Amin Kharaz, Sajjad Arshad, Collin Mulliner, William Robertson, and Engin Kirda	USENIX Security	UNVEIL automatically generates an artificial user environment, and detects when ransomware interacts with user data. In parallel, the approach tracks changes to the systems desktop that indicate ransomware-like behavior.
R-locker: thwarting ransomware action through a honeyfile-based approach	Gomez-Hernandez, JA and Alvarez-Gonzalez, L and Garca-Teodoro	Elsevier	This paper presents a novel approach intended not just to early detect ransomware but to completly thwart its action. For that,a set of honeyfiles are deployed around the target environment in order to catch the ransomware. In addition to frustrate its action, our honeyfile solution is able to automatically launch countermeasures to solve the infection. Moreover, as it does not require previous training or knowledge, the approach allows fighting against unknown, zero-day ransomware related attacks. As a proof of concept, we have developed the approach for Linux platforms called as R-Locker. [1]

ICLDSafe: An Efficient File Backup System in Cloud Storage against Ransomware,	Yun, Joobeom and Hur, Junbeom and Shin, Youngjoo and Koo, Dongyoung	The Institute of Electronics, Information and Communication Engineers	Ransomware becomes more and more threatening nowadays. In this paper, we propose CLDSafe, a novel and efficient file backup system against ransomware. After our system measures file similarities between a new file on the client and an old file on the server, the old file on the server is backed up securely when the new file is changed substantially. And then, only authenticated users can restore the backup files by using challenge-response mechanism. [2]
Detecting ransomware with honeypot techniques	Moore, Chris	IEEE	Attacks of Ransomware are increasing; this form of malwar bypasses many technical solutions by leveraging social engineering methods. This means established methods of perimeter defence need to be supplemented with additional systems. Honeypots are bogus computer resources deployed by network administrators to act as decoy computers and detect any illicit access. This study investigated whether a honeypot folder could be created and monitored for changes. The investigations determined a suitable method to detect changes to this area.[3]

Using software-defined networking for ransomware mitigation: the case of cryptowall	Cabaj, Krzysztof and Mazurczyk, Wojciech	IEEE Network	Currently, different forms of ransomware are increasingly threatening Internet users. Modern ransomware encrypts important user data, and it is only possible to recover it once a ransom has been paid. In this article we show how software-defined networking can be utilized to improve ransomware mitigation.Then we describe the design of an SDN-based system, implemented using OpenFlow, that facilitates a timely reaction to this threat, and is a crucial factor in the case of crypto ransomware. What is important is that such a design does not significantly affect overall network performance. Experimental results confirm that the proposed approach is feasible and efficient.[4]

Causality reasoning about network events for detecting stealthy malware activities,	Zhang, Hao and Yao, Danfeng Daphne and Ramakrishnan, Naren and Zhang, Zhibin	Elsevier	We propose to discover the triggering relations on network requests and leverage the structural information to identify stealthy malware activities that cannot be attributed to a legitimate cause.We design and compare rule- and learning-based methods to infer the triggering relations on network data.We further introduce a user-intention based security policy for pinpointing stealthy malware activities based on a triggering relation graph. [5]
The Detection of 8 Type Malware botnet using Hybrid Malware Analysis in Executable File Windows Operating Systems	Satrya, Gandeva B and Cahyani, Niken DW and Andreta, Ritchie F	International Conference on Electronic Commerce 2015	To distinguish and recognize a malware botnet required malware investigation on Windows executable record. Be that as it may, by and large talking there are two methods in malware examination. That is static investigation and dynamic examination. By consolidating both the aftereffects of static investigation, dynamic examination can create information for distinguishing malware botnet in the executable records of Windows working framework that are Herpestnet, Ann Loader, mbot, Vertexnet, Athena, Elite Loader, Gbot, and Cythosia.[6]

Table 2.1: Survey of Research Papers

Chapter 3

Working of Ransomware

In lock screen ransomware, it doesn't scramble the individual records, it simply locks the screen and requests installment. While, Crypto ransomware encodes the individual documents/data. In this sort of ransomware, records are encrypted and after encryption, client is educated that his information is encoded and won't be decrypted until an ransom amount is paid. Investigation demonstrates that Malware utilizes AES+RSA Encryption. Despite the fact that RSA utilizes asymmetric keys; one is open which is available by outside gathering and the there is private key, just kept by the client. While AES is a symmetric key cryptography, which has just a single key i.e one key uses for both encryption and decoding. AES key is utilized for document encryption Encrypted records are utilized for putting away AES key for decoding. A RSA open key is encoded with this AES key it is possible that we can state , for decoding there is a need of a private key. Three type of ransomware are:-

- Private Key cryptosystem Ransomware
- Public key cryptosystem Ransomware (PuCR)
- Hybrid cryptosystem Ransomware (HCR)

In PrCR, the perspective of the Ransomware author and the perspective of the malware investigator is symmetric. For making the view unbalanced, the key must be expelled From the malware investigator's view, however it is conceivable to recoup the key again By brute force attack or reverse engineering. Be that as it may, the way that everything is obvious to the investigator, is the significant disservice of utilizing this cryptosystem.[2] In PuCR, there is a couple of keys known as Public key and Private key or we state encryption key and decoding key individually. Public key is utilized for encrypting data/information on the casualty machine, while private key is kept by the malware in hidden way. In this way, it would not be feasible for the malware investigator to recognize this private key and this match of key is produced just once, so the information is unscrambled just when casualty is consented to pay the payoff in return of the private key. Be that as it may, this approach likewise has such a variety of disadvantages ,in this, malware attacker can't free one victime at once, he needs to hold everybody until all victims pay their ransom payment in light of the fact that in the event that he liberates one victim, that victim could uncover the private key, it can be overcomed if PuCR produces different key sets. Another downside is that the symmetric encryption plans are substantially speedier than unbalanced encryption plans. To overcome previously mentioned disadvantages, HCR is produced. For this situation, a couple of asymmetric keys are produced again and public key is place in malware payload. Be that as it may, for the information encryption handle an irregular secret key is created on every casualty machine, and the hostage information are encoded utilizing this keyand a quick symmetric cipher. The irregular produced secret key is encrypted utilizing public key and just put away along these lines. For this situation the enemy is not required to unveil his private key. The malware attacker requests the ransom and for de-crypting, the cipher content of the irregular secret key is adequate3.1. He then decrypts the mystery key utilizing the private key and sends it back to the casualty. In this method, with a high likelihood every casualty has an exceptional key, thus distributing of the unscrambling key is of no assistance to other victims.There are many file encrypting ransomwares, such as:-

- Simple Locker
- CryptoLocker
- CTB-Locker
- Torrent Locker

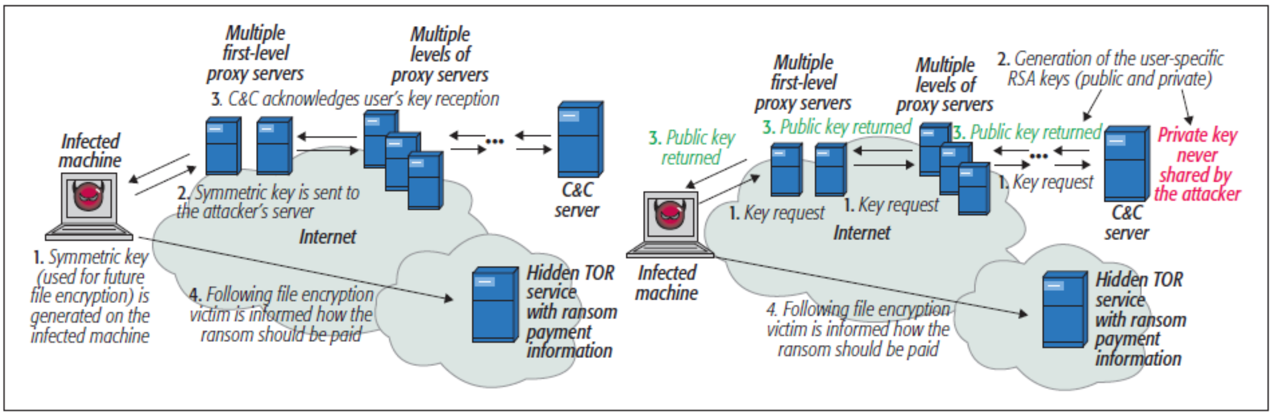

Multiple first-level proxy servers
Multiple levels of proxy servers
3. C&C acknowledges user's key reception
Infected machine
2. Symmetric key is sent to the attacker's server
C&C server
Internet
1. Symmetric key (used for future file encryption) is generated on the infected machine
4. Following file encryption victim is informed how the ransom should be paid
Hidden TOR service with ransom payment information
Multiple first-level proxy servers
Multiple levels of proxy servers
2. Generation of the user-specific RSA keys (public and private)
3. Public key returned
3. Public key returned
3. Public key returned
Private key never shared by the attacker
1. Key request
1. Key request
1. Key request
C&C server
Internet
Infected machine
4. Following file encryption victim is informed how the ransom should be paid
Hidden TOR service with ransom payment information

Figure 3.1: Ransomware CC Server Connection

Chapter 4

Proposed System

Paying the payoff does not take care of the issue on the grounds that there isn't guarantee neither to recoup the information nor to endure again the extortion to keep paying!.
As per the typical task did by ransomware, regular particular occasions allude to file system activity are as follows:

- Increasing number of documents with surely understood expansions like, .locky.
- Modificacion of particular records like PIPE.
- Execution of exceptional charges such as vssadmin, to clean shadow copy.
- To Modify the MBR (Master Boot Record), to directly boot malware screen.

Other ransomware-particular occasions are identified with API calls. For instance, countless ransomware tests utilize capacities like CreateDesktop to bolt the casualties work area by making another one and making it tireless. Also, impairing some console alternate ways will keep the casualty bypassing blocking. On account of crypto ransomware malware, the utilization of standard framework capacities like CryptEncrypt is basic to encode records. Remorsefully, this can be effectively skirted by aggressors through the improvement of their own cryptosystems.

Thinking about impacts of ransomware, we introduce a general useful procedure went for impeding crypto-ransomware activity. It ought to be lightweight while precise and effective in crushing the Zero Day Attack. In light of this, and as a proof of idea, we will implement R-Locker, for Windows OS, similar to R-Locker Implementation done on

Linux OS.

Crypto-ransomware activity depends on examining the tainted machine's filesytem to discover records, either aimlessly or specifically as per particular document expansions , and get to them to encode the data. In view of this general conduct, we propose as a novel hostile to ransomware answer for make a honeyfile planned to fill in as a trap to catch the malware. Such a proposition will exhibit the accompanying highlights:
The ransomware test will be conclusively blocked while getting to the honeyfile, with the goal that whatever remains of the framework will stay undamaged and locking the ransomware, the vindictive occasion ought to be appropriately advised and additionally a countermeasure naturally sent to settle the danger in Figure4.2.

The above system compares to the useful design appeared in Figure 4.1. Such an operational technique is calculated and ought to be free on the particular target stage or OS considered (Windows, Unix, iOS, and so forth). Not withstanding the past wanted hostile to recover activity, some different requests ought to be fulfilled with a specific end goal to get an adaptable, usable and, in that capacity, substantial answer for genuine situations.[4]

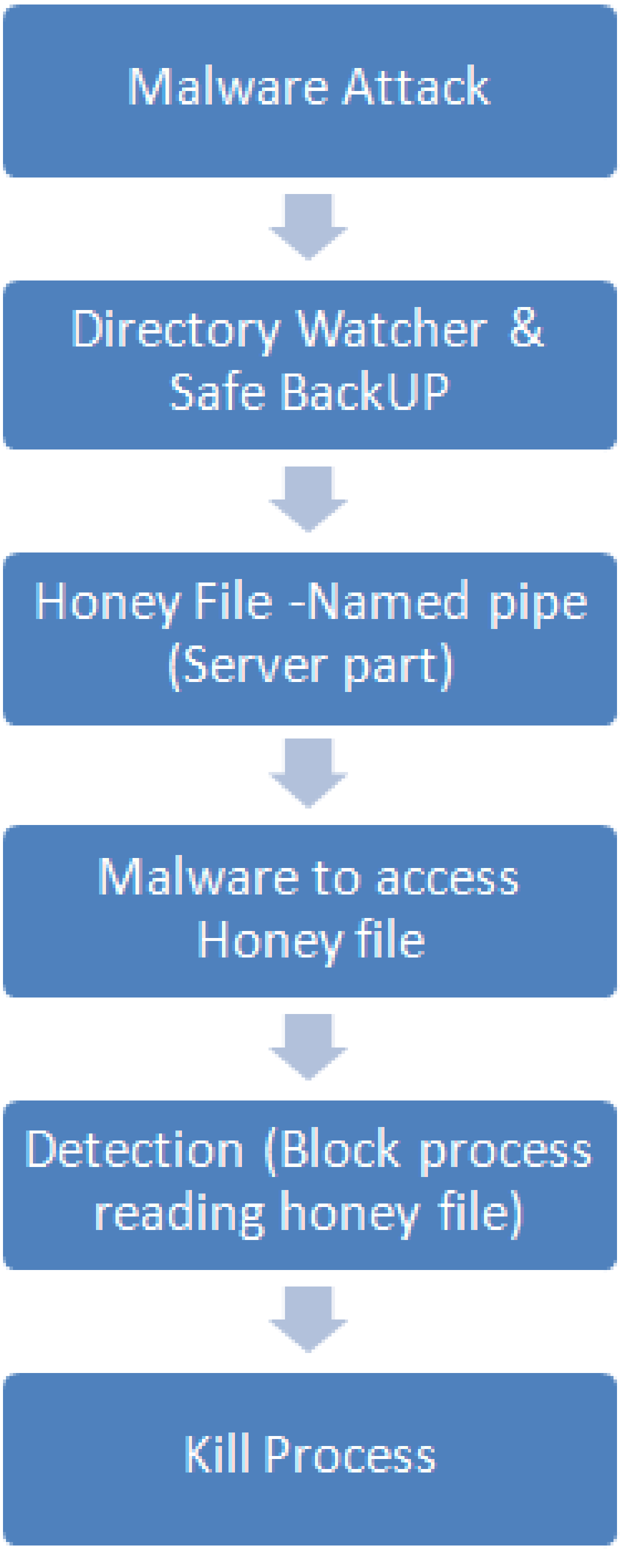

Malware Attack
Directory Watcher & Safe BackUP
Honey File -Named pipe (Server part)
Malware to access Honey file
Detection (Block process reading honey file)
Kill Process

Figure 4.1: Flow Diagram of The System

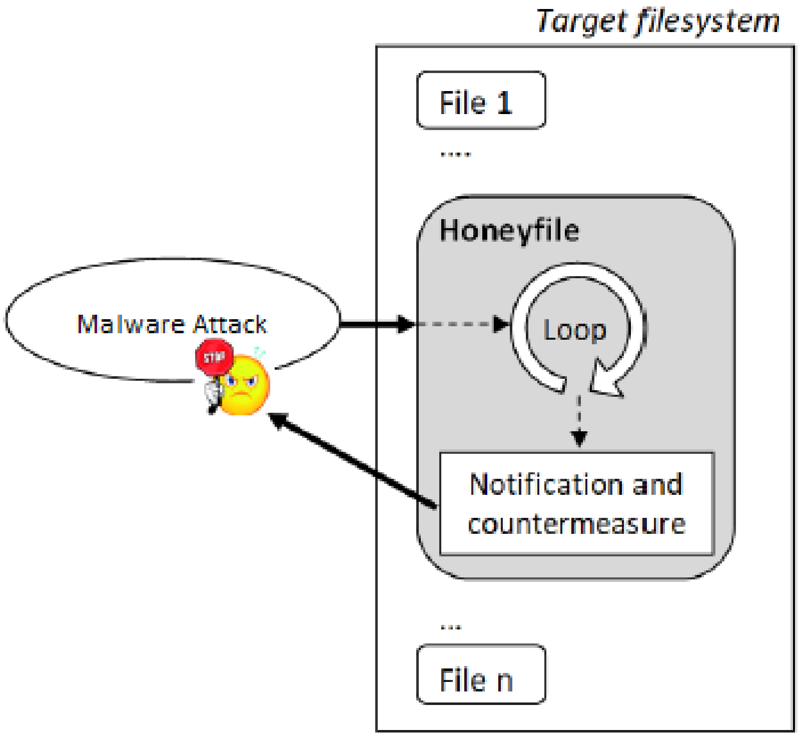


Figure 4.2: R-Locker Working[1]
g

Chapter 5

Implementation And Results

A basic and exquisite answer for accomplishing our objectives, both, while fulfilling prerequisites. R-Locker is built up to make the arrangement and utilize named pipe or FIFOs. A FIFO file is a pipe with a name into the filesystem, and with two exceptionally intriguing and helpful properties for our motivation because of such a double nature

- It initially makes a named pipe by utilizing the capacity CreateNamedPipe(). This will be our focal honeyfile or to trap the malware.
- Some bytes are written on client side. The bytes ought to be not the same as EndOfFile bytes and the quantity of them will rely upon the particular framework. [1]

5.1 Tools and Technology

Programming Language:- Java C

Library/ Platform:- POI Library

5.2 System Configuration

Operating System:- Windows 8

OS Type:- 64-bit operating system

Processor:- Intel(R) Core(TM) i5

RAM:- 4 GB

5.3 R-Locker

Under normal operation of the environment, and with R-Locker installed and running, however as we could not detect attack as Named Pipe could not be mounted on Windows-8 system as it does not gets mounted to Named Pipe File System in Figure 5.1. However this could also detect if more than 50 files have been modified in 1 minute. Also incremental backup is implemented which does backup of only updated file.

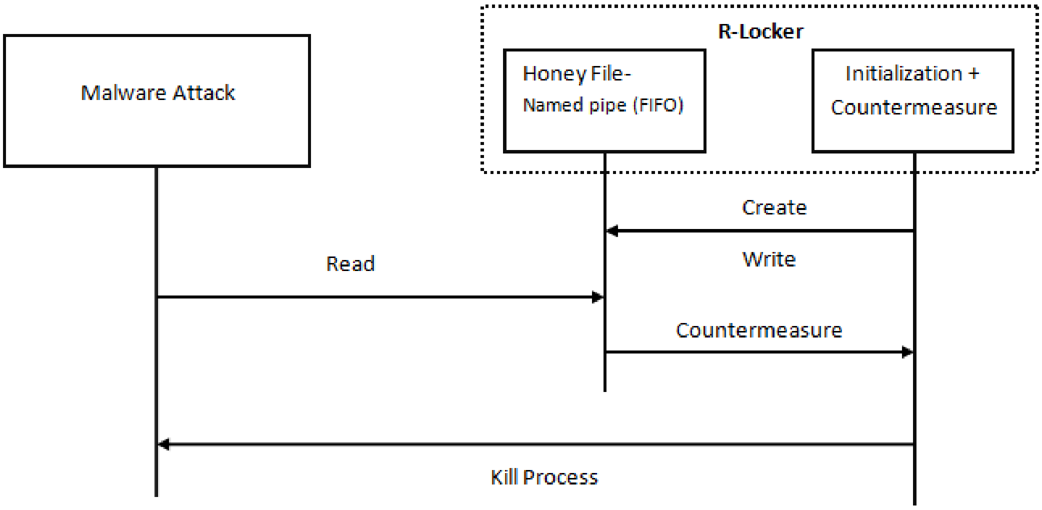
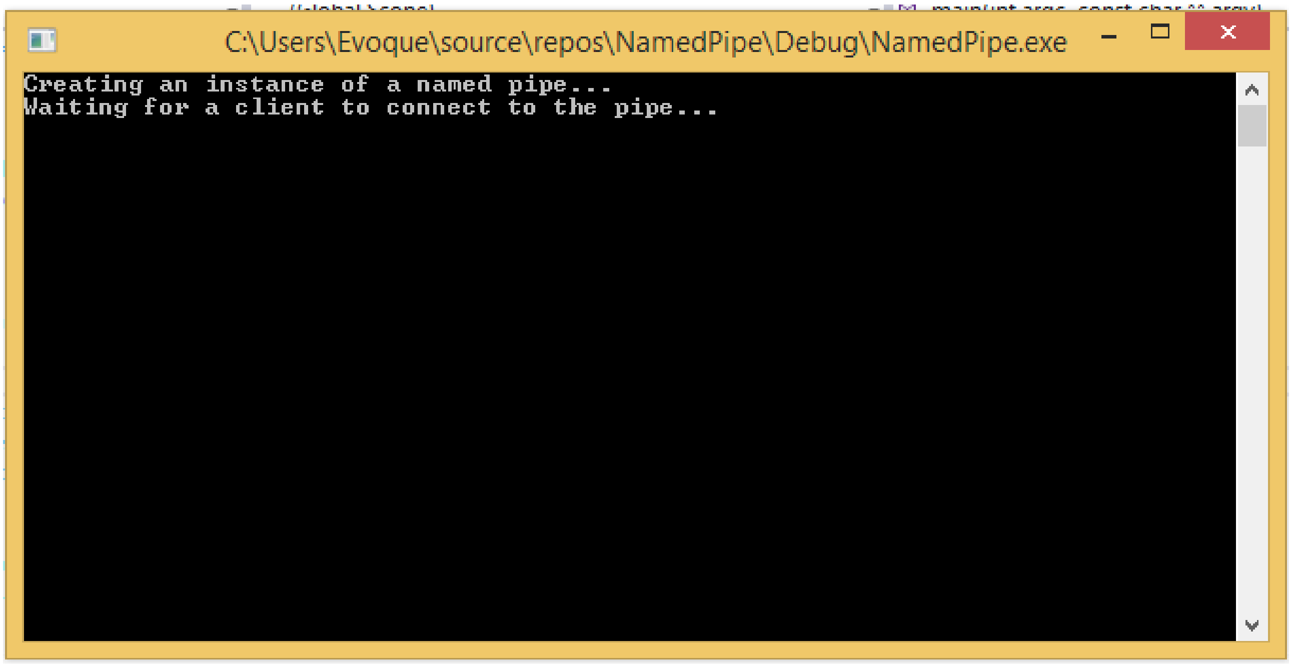


Figure 5.1: R Locker Working [1]

Honey File or Named pipe Advantage

- Named pipes are FIFO in nature
- Used for IPC
- Lower in size and consumption
- Can be easily created and deleted
- Work like client-server in windows

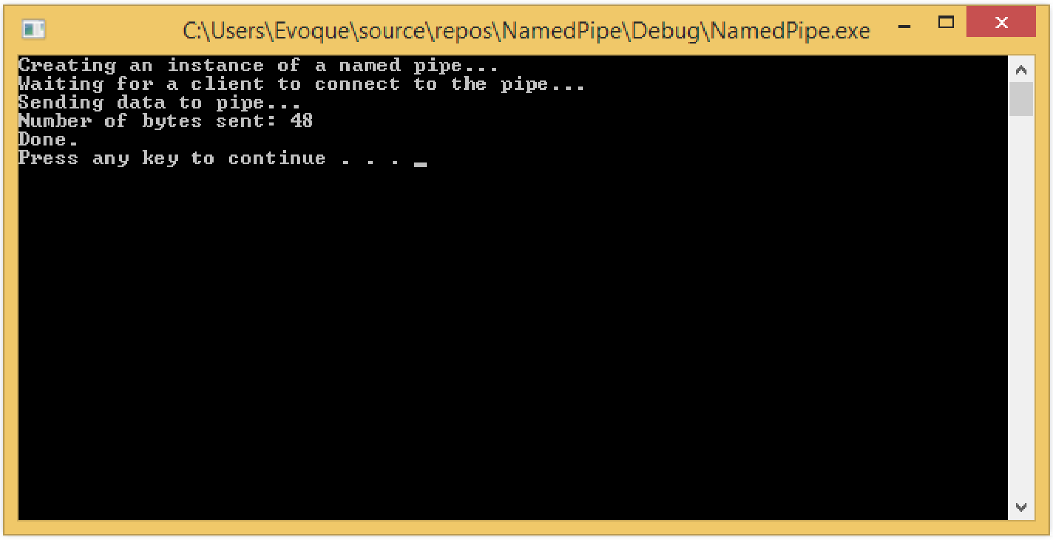
Named pipe server creation refer Figure 5.2



```
Creating an instance of a named pipe...
Waiting for a client to connect to the pipe...
```

Figure 5.2: Named Pipe Server Creation

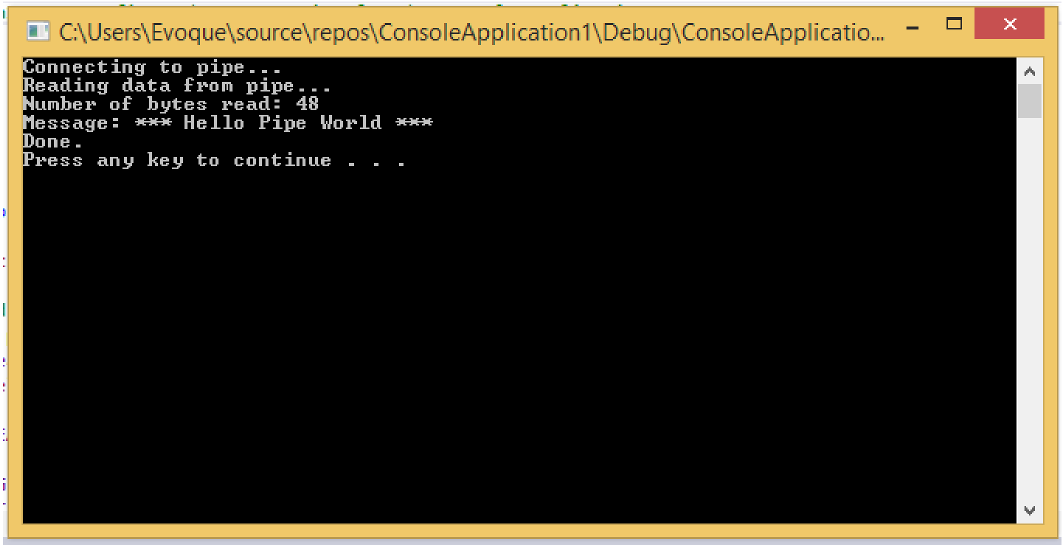
Named pipe server to client connection refer Figure 5.3



```
Creating an instance of a named pipe...
Waiting for a client to connect to the pipe...
Sending data to pipe...
Number of bytes sent: 48
Done.
Press any key to continue . . . _
```

Figure 5.3: Named Pipe Server

Named pipe server to client connection refer Figure 5.4



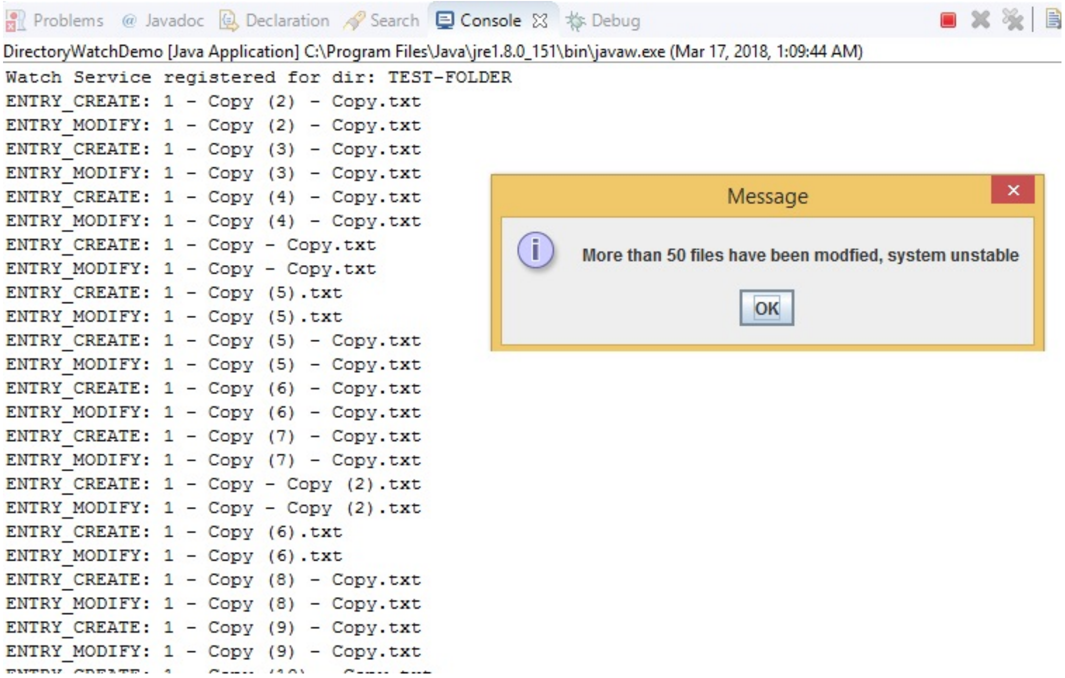
C:\Users\Evoque\source\repos\ConsoleApplication1\Debug\ConsoleApplicatio...

```
Connecting to pipe...
Reading data from pipe...
Number of bytes read: 48
Message: *** Hello Pipe World ***
Done.
Press any key to continue . . .
```

Figure 5.4: Named Pipe Client

R-Locker Additional utility

In this R-Locker utility a directory is watched and if more than 50 files has been modified then this raises alert for the same refer Figure 5.5.



DirectoryWatchDemo [Java Application] C:\Program Files\Java\jre1.8.0_151\bin\javaw.exe (Mar 17, 2018, 1:09:44 AM)

```
Watch Service registered for dir: TEST-FOLDER
ENTRY_CREATE: 1 - Copy (2) - Copy.txt
ENTRY_MODIFY: 1 - Copy (2) - Copy.txt
ENTRY_CREATE: 1 - Copy (3) - Copy.txt
ENTRY_MODIFY: 1 - Copy (3) - Copy.txt
ENTRY_CREATE: 1 - Copy (4) - Copy.txt
ENTRY_MODIFY: 1 - Copy (4) - Copy.txt
ENTRY_CREATE: 1 - Copy - Copy.txt
ENTRY_MODIFY: 1 - Copy - Copy.txt
ENTRY_CREATE: 1 - Copy (5).txt
ENTRY_MODIFY: 1 - Copy (5).txt
ENTRY_CREATE: 1 - Copy (5) - Copy.txt
ENTRY_MODIFY: 1 - Copy (5) - Copy.txt
ENTRY_CREATE: 1 - Copy (6) - Copy.txt
ENTRY_MODIFY: 1 - Copy (6) - Copy.txt
ENTRY_CREATE: 1 - Copy (7) - Copy.txt
ENTRY_MODIFY: 1 - Copy (7) - Copy.txt
ENTRY_CREATE: 1 - Copy - Copy (2).txt
ENTRY_MODIFY: 1 - Copy - Copy (2).txt
ENTRY_CREATE: 1 - Copy (6).txt
ENTRY_MODIFY: 1 - Copy (6).txt
ENTRY_CREATE: 1 - Copy (8) - Copy.txt
ENTRY_MODIFY: 1 - Copy (8) - Copy.txt
ENTRY_CREATE: 1 - Copy (9) - Copy.txt
ENTRY_MODIFY: 1 - Copy (9) - Copy.txt
```

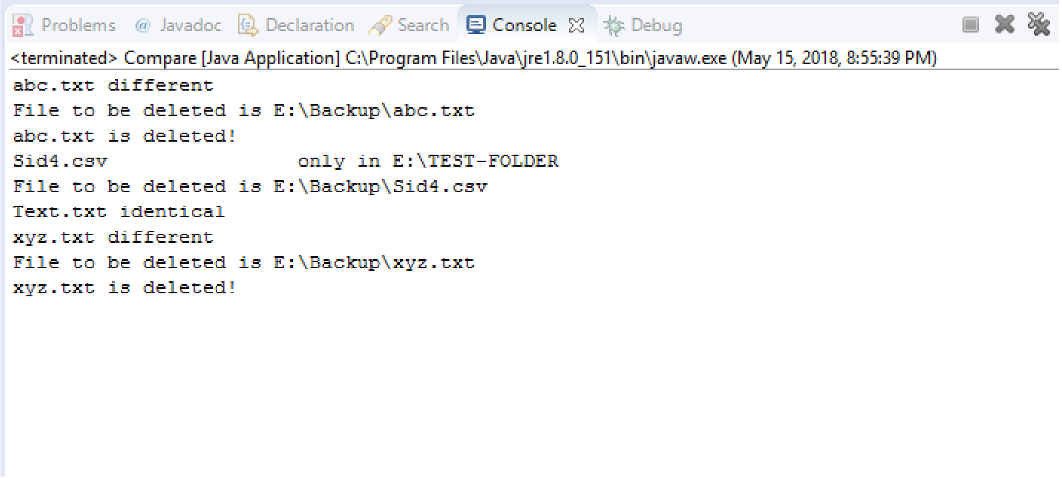
Message

More than 50 files have been modfied, system unstable

Figure 5.5: Directory Watcher

Incremental Backup Technique

This backup technique would get the hash of both the files and compare the same and which show if both of these match. If the hash of these files does not match then same file is backed up to the Backup Folder refer Figure5.6. [2]



```
<terminated> Compare [Java Application] C:\Program Files\Java\jre1.8.0_151\bin\javaw.exe (May 15, 2018, 8:55:39 PM)
abc.txt different
File to be deleted is E:\Backup\abc.txt
abc.txt is deleted!
Sid4.csv                only in E:\TEST-FOLDER
File to be deleted is E:\Backup\Sid4.csv
Text.txt identical
xyz.txt different
File to be deleted is E:\Backup\xyz.txt
xyz.txt is deleted!
```

Figure 5.6: Incremental Safe Backup

Chapter 6

Conclusion and Future Work

A general strategy proposed to foil crypto-ransomware activity is presented here. It depends on the arrangement of a honey file structure to hinder the payoff when it gets to a trap document, in this manner permitting to protect whatever remains of the information on the framework. In addition, while the payoff is blocked, it is attractive to consequently dispatch a countermeasure planned to kill the process from the system. [3]As a proof of idea, R-Locker has been implemented on Windows stages by making utilization of named pipes or FIFOs. However as further work, we are dealing with enhancing our present execution in a portion of the perspectives, specifically, for Windows. Despite the fact that the general honeyfile arrangement is pertinent to the two kinds of stages, some particular perspectives ought to be deliberately routed to give real arrangements. Specifically, named pipes are excluded into the typical file system space in Windows. In addition to threat detection technique we have also implemented incremental safe backup technique this helps us to reduce latency.

Future Works:-

- Mounting Named Pipe to Windows file system.
- Backing up on Cloud Drive.
- Integrating Threat Detection with Virus Total Scanner.

Bibliography

[1] J. Gomez-Hernandez, L. Alvarez-Gonzalez, and Garca-Teodoro, “R-locker: Thwarting ransomware action through a honeyfile-based approach,” *Computers and Security*, vol. 73, pp. 389–398, 2018.

[2] J. Yun, J. Hur, Y. Shin, and D. Koo, “Cldsafe: An efficient file backup system in cloud storage against ransomware,” *IEICE TRANSACTIONS on Information and Systems*, vol. 100, no. 9, pp. 2228–2231, 2017.

[3] C. Moore, “Detecting ransomware with honeypot techniques,” in *Cybersecurity and Cyberforensics Conference (CCC), 2016*, pp. 77–81, IEEE, 2016.

[4] K. Cabaj and W. Mazurczyk, “Using software-defined networking for ransomware mitigation: the case of cryptowall,” *IEEE Network*, vol. 30, no. 6, pp. 14–20, 2016.

[5] H. Zhang, D. D. Yao, N. Ramakrishnan, and Z. Zhang, “Causality reasoning about network events for detecting stealthy malware activities,” *computers & security*, vol. 58, pp. 180–198, 2016.

[6] G. B. Satrya, N. D. Cahyani, and R. F. Andreta, “The detection of 8 type malware botnet using hybrid malware analysis in executable file windows operating systems,” in *Proceedings of the 17th International Conference on Electronic Commerce 2015*, p. 5, ACM, 2015.

[7] L. J. G. Villalba, A. L. S. Orozco, and J. M. Vidal, “Advanced payload analyzer preprocessor,” *Future Generation Computer Systems*, vol. 76, pp. 474–485, 2017.

[8] B. A. S. Al-rimy, M. A. Maarof, and S. Z. M. Shaid, “Ransomware threat success factors, taxonomy, and countermeasures: a survey and research directions,” *Computers & Security*, 2018.

[9] D. Štitilis, P. Pakutinskas, M. Laurinaitis, and I. M.-v. de Castel, "A model for the national cyber security strategy. the lithuanian case.," *Journal of Security & Sustainability Issues*, vol. 6, no. 3, 2017.

[10] R. Koizumi and R. Sasaki, "Study on countermeasures using mitigation software against vulnerability attacks," in *Cyber Security, Cyber Warfare, and Digital Forensic (CyberSec), 2015 Fourth International Conference on*, pp. 28–33, IEEE, 2015.

[11] M. M. Ahmadian and H. R. Shahriari, "2entfox: A framework for high survivable ransomwares detection," in *Information Security and Cryptology (ISCISC), 2016 13th International Iranian Society of Cryptology Conference on*, pp. 79–84, IEEE, 2016.

[12] M. Weckstén, J. Frick, A. Sjöström, and E. Järpe, "A novel method for recovery from crypto ransomware infections," in *Computer and Communications (ICCC), 2016 2nd IEEE International Conference on*, pp. 1354–1358, IEEE, 2016.

[13] F. Zhang and Y. Ma, "Using irp with a novel artificial immune algorithm for windows malicious executables detection," in *Progress in Informatics and Computing (PIC), 2016 International Conference on*, pp. 610–616, IEEE, 2016.

[14] A. Zahra and M. A. Shah, "Iot based ransomware growth rate evaluation and detection using command and control blacklisting," in *Automation and Computing (ICAC), 2017 23rd International Conference on*, pp. 1–6, IEEE, 2017.

[15] A. Jerlin and J. Chinnappan, "Esaa: Efficient sequence alignment algorithm for dynamic malware analysis in windows executable using api call sequence," *DNA sequence*, vol. 291, 2017.

[16] M. Alazab, S. Venkatraman, P. Watters, and M. Alazab, "Zero-day malware detection based on supervised learning algorithms of api call signatures," in *Proceedings of the Ninth Australasian Data Mining Conference-Volume 121*, pp. 171–182, Australian Computer Society, Inc., 2011.

[17] R. Kaur and M. Singh, "A hybrid real-time zero-day attack detection and analysis system," *International Journal of Computer Network and Information Security*, vol. 7, no. 9, p. 19, 2015.

[18] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues in Information Warfare & Security Research*, vol. 1, no. 1, p. 80, 2011.

[19] A. Kharraz, S. Arshad, C. Mulliner, W. K. Robertson, and E. Kirda, "Unveil: A large-scale, automated approach to detecting ransomware.," in *USENIX Security Symposium*, pp. 757–772, 2016.