RANSOMWARE ANALYSIS BASED ON THE SURFACE, RUNTIME AND STATIC CODE METHOD

[1] LULUK USMAN, [2] YUDI PRAYUDI, [3]IMAM RIADI

[1,2] Department of Informatics, Universitas Islam Indonesia, Jln. Kaliurang km.14.5, Yogyakarta, Indonesia
[3] Ahmad Dahlan University, Jln. Prof.Dr.Soepomo, Janturan, Yogyakarta, Indonesia
E-mail : [1] lulukusman@gmail.com, [2]prayudi@uii.ac.id, [3]imam.riadi@is.uad.ac.id

ABSTRACT

Ransomware is one of the latest malware in recent years that can infect computers and smartphones. The malware is able to encrypt the files inside the computer or smartphone, thus prevents the users (victims) from accessing their system. In addition, the victims will be asked to pay the ransom through certain online payment methods to get a decrypt key. Due to the latest development of ransomware variants, a solution is required to prevent the malware attack. This study analyzes the cryptolockers ransomware which utilize three method such as surface, runtime and static code method. The result provided the detail characteristics of ransomware through three aforementioned methods as well as the solution to prevent the attack.

Keywords: *Ransomware, Surface, Runtime, Static Code.*

1. INTRODUCTION

As the increasing of technological developments, the cyber threats on computers have been increasing as well. One of the latest malware which has been found in the last few years is Ransomware. This malware has the ability to paralyze the computer data thus unable users to access their system. The purpose of the malware is to squeeze out the infected computer software and request for payment so that the computer can be normal as before [1].

Dell research team, SecureWorks Counter Threat Unit (TM) (CTU) has analyzed the presence of malware file-encrypting which are distributed over the Internet in late February 2014 and known as Cryptolocker. Although it began to be known in the first quarter of 2014, but it has actually been distributed since at least early November 2013. The CTU researchers have assumed that Cryptolocker would become the biggest and most destructive ransomware on the internet [2] and it is proven until now that the cryptolocker is still releasing new variants.

The encryption technique used by Cryptolocker is RSA 2048 and it is utilized largely by technology companies such as Yahoo, Google, Facebook; financial and e-commerce companies to protect financial and other important transactions data. The Cryptolocker used RSA 2048 to encrypt the data inside computer victims and control its decryption key (Private Key). Even though the ransom money has been paid by victims, there is no guarantee that the decryption process will be success.

The goal of our study is to utilize three-stage approach such as surface, runtime and static-code. By analyzing these ransomware in detail, the evidence of digital crime can be collected and the future attack can be prevented.

2. RELATED WORK

Distler [3] in his research has explained the approach in malware analysis: static (code) and dynamic (behavioral). Static is the actual viewing of code in order to get a better understanding of the malware, while the dynamic is to analysis the changes when the malware is executed.

The related research has been done by Rick Flores on win32 malware, kryptic, by utilizing the static analysis [4]. In current study, it is started with performing the hashing on malware, and then followed by collecting information from malware as well as detecting the server which has direct communication to the malware. In this study only utilized static analysis method so that the information obtained is still not completed.

Konstantinou [5] in his study of Metamorphic virus analysis, describes the transformation of metamorphic viruses. First, the code was prepared to produce the new code. Second, the engine needs to decode the information required to perform the transformations. Third, in order for the metamorphic transformations to work correctly, certain information must be available. Fourth, transforming the code into equivalent code, and the last step is to attach the new generation of the virus to a new host file.

Malhotra [6] has conducted a study related to mobile Malware and several techniques are used for malware detection. Two main techniques are usually utilized by researchers, signature-based and anomaly-based. In the signature-based techniques, sets instruction patterns are studied and analyzed, while in anomaly-based the unusual activities can be detected. The result has suggested that the signature based techniques can be enhanced using DNA matching techniques applied in other domains.

Nugroho & Prayudi [7] have suggested several processes on analyzing the malware such as determining the SOP, defining malware, malware analysis, determining the goals of basic malware analysis and reverse engineering of malware (assembly, disassembly, debugging). Currently, the Reverse Engineering is one of the solutions that can be used to analyze the malware. Reverse Engineering in malware analysis is used to extract the hidden data which are considered as important information inside malware.

Yusirwan et al [8] have conducted the analysis of malware based on static and dynamic analysis and the TT.exe is used as ransomware sample. The current study has concluded that Malware TT.exe is a trojan type malware; and windows 7 and windows 8 are the main target. When the computers are infected by this malware, the high RAM usage is used by the malware and infected other programs in computer victim. The current study has successfully extracted the malware characteristics, how it works and the impact which are affected from the attack.

Our study focus on ransomware cryptlocker with variants DOMSTOLSPROCESS_17924.exe. This cryptolocker has been detected by the end of 2016. To the best of our knowledge, no research has been done on analyzing aforementioned malware based on the surface, runtime and static method. Thus by analyzing this cryptolocker in detail, the future attack can be prevented.

3. RANSOMWARE ANALYSIS

In this part, the sample of ransomware is collected. Once the sample is collected and controlled environment is prepared, the detail analysis can be presented.

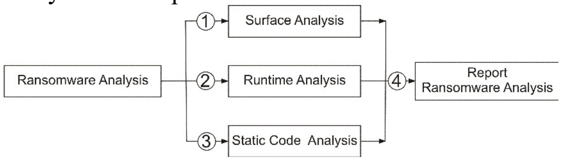


Figure 1: Ransomware analysis method

Figure 1 describes the step by step of analyzing ransomware such as Surface, Runtime and Static Code Analysis. In the last part, the full report of aforementioned techniques can be presented.

3.1 Surface Analysis

This technique is done by observing the process generated by the Ransomware. Some examples of ransomware processes such as creating the file/directory, deletions, changing the file name and other [9]. This technique is trying to detect Ransomware with basic observation without executing the ransomware. Understanding ransomware characteristics can be done by using special software, such as: HashTab and digest.exe (Hash Analysis), TrID (an Analysis), BinText and URstrings.exe (String Analysis), HxD (Binary Editor), CFF Explorer (Pack Analysis), and 7zip (Archiver).

3.2 Runtime Analysis

This technique is done by observing processes that occured in the system, particularly on the process that is directly related to the operating system. This process required special software to analyze the ransomeware, since most system processes run in the background [9]. In runtime analysis, the ransomware should be executed in a controlled environment and its behavior and impact are observed.

Once the Ransomware is executed, the impact on the system can be analyzed. The tool or software to analyze the ransomware such as: Process Explorer, Regshot, Wireshark, TCPView, Process Monitor, Autoruns, FUNdelete, Streams/ADSSpy, and others. These applications are executed on the client side; while for the server side the softwares such as FakeDNS, netcat/ncat and tcpdump/tshark can be utilized.

3.3 Static Code Analysis

The last analysis technique in this study, static code requires the ability of understanding the assembly programming. In this process, the malware code is broken down into its machine instructions (disassembling) and the instructions are analyzed. This is the most effective technique for determining what the malware actually does [9].

This requires more skill and expertise than the other analysis techniques. There many software which are utilized for the static code analysis such as: IDA Pro (Disassembler); Hex-Rays, Reflector, .NET and VB Decompiler (Decompiler); Msdn

Library, Google (Library); OllyDbg, Immunity Debugger, WinDbg/Syser (Debugger); HxD, WinHex, 010editor (Hex Editor); Python, Linux/Cygwin/MSYS Shell (Others).

3.4 Ransomware Analysis Report

Once the process of analyzing ransomware based on surface, runtime, static code are completed, the report related with the characteristics of Ransomware can be presented. In this final report, the recommendations on preventing the ransomware will be presented. In addition, the characteristics of Ransomware also will be presented, so that the researchers can estimate the damage caused by Ransomware.

4. PRACTICAL RANSOMWARE ANALYSIS TECHNIQUE

This section will describe some of the detail techniques which are used for analyzing Ransomware.

4.1 Packed and Obfuscated Program

Packed or repacked malware is a method to change the malware with compressed or encryption so that the malware will be difficult to be recognized by antivirus and/or malware researchers [10]. Whereas obfuscated is a method to make the source code of a malware becomes difficult to be understood. Obfuscated normally used by network programmers for security reason, so that the program will be difficult to be hacked. However, these technique is used by malware creator to make malware becomes more difficult to be detected and analyzed by malware researchers [11].

4.2 Deobfuscated

Deobfuscated is a method to restore the programs that had previously been obfuscated. By doing a deobfuscated step, the language of a program that had previously been scrambled can be restored as before, so that it will make easier for researchers to conduct analysis on malware [12].

4.3 Packet Capture

Packet capture is the process of recording the data packets which passing through computer networks. In computer networks, for example on HTTP protocol, there was communication between client and server. Communication requires both data exchange, so that we can record, filter, analyze, diverse the data packets from a variety of protocol by using special application.

4.4 Debugging

Debugging is an activity to find out the error in a program.These errors are usually called bugs. The bug can be either error logic, syntax errors, or the errors on accessing the device that are not allowed by operating system [13].

5. TOOL FOR ANALYSIS

In this study, several tools based on surface, runtime and static code analysis are presented. Tool for surface analysis are described in Table 1.

Table 1. Tool used for Surface Analysis

Surface Analysis Tools	Describe
Virustotal.com	A free online service that analyzes the files and URLs enabling the identification of viruses, worms, trojans and other kinds of malicious content detected by antivirus engines and website scanners.
QuickHash	A utility to quickly display the MD5, SHA1, SHA256, SHA384, SHA512, (and SHA3 in v2.x) hashes of any selected file, and optionally compare the hashes with any hash string
PEiD v0.95	An intuitive application that relies on its user-friendly interface to detect packers, cryptors and compilers found in PE executable files
Exeinfo PE v0.0.3.6	A software that can be used to view various information on any executable file.
MASTIFF Online	A static analysis framework that automates the process of extracting key characteristics from a number of different file formats.
ThreatExpert	An advanced automated threat analysis system designed to analyze and report the behavior of computer viruses, worms, trojans, adware, spyware, and other security-related risks in a fully automated mode.

For analyzing the ransomware based on runtime analysis, the tool which can be utilized can be seen in Table 2.

Table 2. Tool used for Runtime Analysis

Runtime Analysis Tool	Describe
Unit Computer	A computer unit used to analyze the Ransomware.
Regshot	An open-source registry compare utility that allows to quickly take a snapshot of registry and then compare it with a second one.
Process Monitor	A monitoring software for Windows that displays real-time system, process/thread and Registry activity.
NetworkMiner	A Network Forensic Analysis Tool for Windows, can be used as a passive network sniffer/packet capturing tool in order to detect operating systems, sessions, hostnames, open ports etc. without putting any traffic on the network
ApateDNS	Tools to Fake DNS Responses for Ransomware Analysis
Wireshark	A free and open source packet analyzer, it allows user to see what's happening on network at a microscopic level.

For analyzing the malware based on static code analysis, the tools which can be utilized are presetend in Table 3.

Table 3. The tool that is used for Static Code Analysis

Static Code Analysis Tool	Describe
Dependency Walker	A free program for Microsoft Windows used to list the imported and exported functions of a portable executable file.
BinText	A file text scanner / extractor that helps find character strings buried in binary files.
OllyDbg	A 32-bit assembler level analysing debugger for Microsoft Windows

This study used cryptolocker as ransomware sample. The detail explanation of the research flow can be seen in Figure 2.

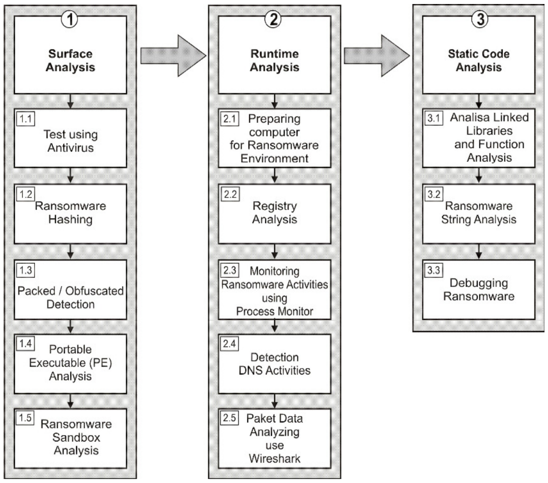

1
Surface Analysis
1.1 Test using Antivirus
1.2 Ransomware Hashing
1.3 Packed / Obfuscated Detection
1.4 Portable Executable (PE) Analysis
1.5 Ransomware Sandbox Analysis
2
Runtime Analysis
2.1 Preparing computer for Ransomware Environment
2.2 Registry Analysis
2.3 Monitoring Ransomware Activities using Process Monitor
2.4 Detection DNS Activities
2.5 Paket Data Analyzing use Wireshark
3
Static Code Analysis
3.1 Analisa Linked Libraries and Function Analysis
3.2 Ransomware String Analysis
3.3 Debugging Ransomware

Figure 2 : Process research of ransomware analysis

Based on the Figure 2, there are several steps that are performed in this study; generally they are grouped into three major groups.

6. RESULT AND ANALYSIS

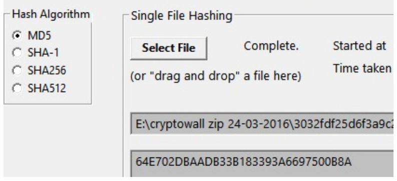
In the first step, the analysis of ransomware is done by surface methods and the tools which are utilized can be seen as follow:

a. Malwr.com
b. Virustotal.com
c. QuickHash-v 2.6.1
d. Exeinfo PE
e. 7zip
f. PEViewer
g. MASTIFFOnline.com
h. ThreatExpert.com

In this study, the ransomware sample was obtained from the website malwr.com. The sample is tested by antivirus and followed by hashing the package of the ransomware.

The ransomware sample is uploaded to virustotal.com and it will be scanned by various antivirus engine and the results can be presented. The results showed that the sample of Cryptolocker is the category of ransomware and 39 out of 57 antivirus engines are succesfully recognize it as a ransomware Trojans.

Hashing is the transformation of a string of characters into a usually shorter fixed-length value or key that represents the original string. The purpose of hashing is to get information the md5 value from the ransomware. To perform the hashing, this study use the free version program QuickHash-v 2.6.1.

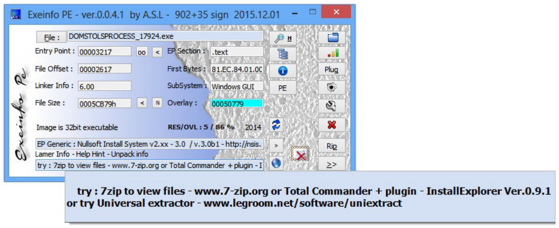


Hash Algorithm
● MD5
○ SHA-1
○ SHA256
○ SHA512

Single File Hashing
Select File
Complete.
Started at
Time taken
(or "drag and drop" a file here)
E:\cryptowall zip 24-03-2016\3032fdf25d6f3a9c
64E702DBAADB33B183393A6697500B8A

Figure 3 : Hashing Ransomware using Quickhash

Figure 3 illustrates the process of hashing with md5. The the value of ransomware hashing can be seen as follow:
Name: Bolletta_64940.EXE or DOMSTOLSPROCESS_17924.exe
MD5:64E702DBAADB33B183393A6697500B8A



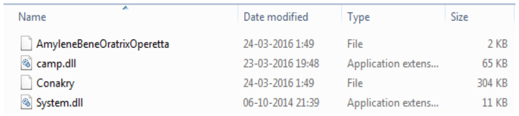
Exeinfo PE - ver.0.0.4.1 by A.S.L - 902+35 sign 2015.12.01
File : DOMSTOLSPROCESS_17924.exe
Entry Point : 00003217
EP Section : .text
File Offset : 00002617
First Bytes : 81.EC.84.01.00
Linker Info : 6.00
SubSystem : Windows GUI
File Size : 0005CB79h
Overlay : 00050779
Image is 32bit executable
RES/OVL : 5 / 86 % 2014
EP Generic : Nullsoft Install System v2.xx - 3.0 / v.3.0b1 - http://nsis.
Lamer Info - Help Hint - Unpack info
try : 7zip to view files - www.7-zip.org or Total Commander + plugin - I

try : 7zip to view files - www.7-zip.org or Total Commander + plugin - InstallExplorer Ver.0.9.1
or try Universal extractor - www.legroom.net/software/uniextract

Figure 4 : Exeinfo has detected packed

Next step is testing using Exeinfo PE to get information about the packed or obfuscated from ransomware. Packed or repacked ransomware is used to upacked the ransomware, thus this process is necessary before doing further analysis.

The result showed that the ransomware was packed with NSIS. The display of the test results with the Exeinfo PE can be seen in Figure 4. It can be seen that the malware has used the ransomware package and to unpack the malware, 7zip can be utilized as can be seen in Figure 5.



Name	Date modified	Type	Size
AmyleneBeneOratrixOperetta	24-03-2016 1:49	File	2 KB
camp.dll	23-03-2016 19:48	Application extens...	65 KB
Conakry	24-03-2016 1:49	File	304 KB
System.dll	06-10-2014 21:39	Application extens...	11 KB

Figure 5 : Result unpacking using 7zip

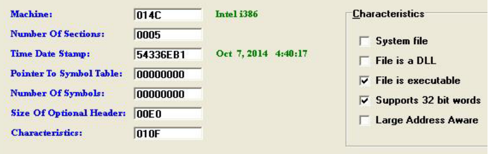
There are several programs that can perform the unpacking process, such as:

a. de4dot
b. GunPacker. V05
c. NETUnpack
d. 7Zip
e. Universal Extractor

The unpacking process showed that there are four files as can be seen in Figure 5. The names are:

a. amyleneBeneOratrixOperetta,
b. Camp.dll,
c. System.dll,
d. Conakry,

The result showed that Camp.dll file is detected as a trojan by antivirus.



Machine: 014C Intel i386
Number Of Sections: 0005
Time Date Stamp: 54336EB1 Oct 7, 2014 4:40:17
Pointer To Symbol Table: 00000000
Number Of Symbols: 00000000
Size Of Optional Header: 00E0
Characteristics: 010F

Characteristics
□ System file
□ File is a DLL
☑ File is executable
☑ Supports 32 bit words
□ Large Address Aware

Figure 6 : Analyzing use PEViewer

Figure 6 shows testing process by utilizing PEViewer and the creation date of malware can be presented. For analyzing the change of registry which is made by ransomware, the Regshot tool can be used. The registry detail of before and after the ransomware is executed, should be captured and compared, thus the change can be analyzed.

At this part, these ransomware are sent to ransomware sandbox and it will be analyzed by engine-MASTIFF. The result of Mastiff ransomware sandbox can be seen as follow:

a. When the ransomware is executed, the mallware will import some DLL files such as kernel32.dll, user32.dll, GDI32.dll, shell32.dll, advapi32.dll, comctl32.dll, ole32.dll and version.dll
b. There is also information about the packed of ransomware and its explains the date of creation.
c. There are no registry change information.

Other sandbox analysis is used, the ThreatExpert and the result can be seen as follow:

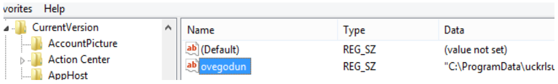
a. There are new processes of creating ughkegib.exe file by the ransomware.
b. Ransomware also made some key in registry.
c. Ransomware communicate with certain IP Address.

For the second part of analysis, the runtime analysis methods is used and controlled environment is needed. One unit computer was used for conducting analysis of ransomware.

Computer configuration:

1. Operating System: Windows 32 bit 8
2. Memory: 1024 MB
3. Hard disk: 80 GB
4. Network: Wifi

After setting up a Controlled Environment, the analysis on windows registry is needed. Regshot is used as the tool for analysis. The regshot is used to capture the registry of before and after the ransomware is executed. Two registry are captured, compared and analyzed the difference.



Name	Type	Data
(Default)	REG_SZ	(value not set)
ovegodun	REG_SZ	"C:\ProgramData\uckrlsu

Figure 7 : Analysis of Registry by using Regshot

Figure 7 shows the information that the ransomware has tried to turn on itself when the computer is restarted.

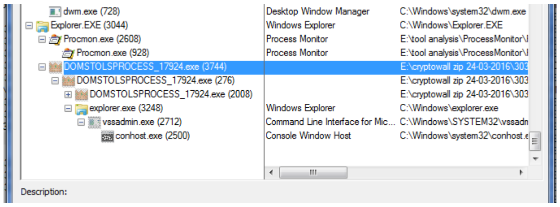
The result showed that the ransomware has created the activities which can be seen as follows:

a. Program schedule is change: HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Schedule\TaskCache\Tasks\{68070BBC-F2DE-4476-95C6-C2ED1ECE3D0F} \ Hash: EC 5 d B0 B1 CC E5 5F AC 8 c 8 c 1B 8E F1 DF E5 2F66 E4 34 BF 7E 96 40 38 29 46 CD D7 CB ED 2B CB

b. Ransomware changes automatic update in windows 7: HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\AutoUpdate\AUOptions: 0x00000004 HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\AutoUpdate\IncludeRecommendedUpdates: 0x1 HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update\CachedAUOptions: 0x00000004

c. Ransomware stops the protection of Windows Defender: HKLM\SOFTWARE\Microsoft\WindowsDefender\Real-Time Protection\DisableRealtimeMonitoring: 0x1

d.Ransomware made itself run automatically when Windows is turned on: HKU\S-1-5-21-3819072523-3082051562-3461999040-1001\Software\Microsoft\Windows\CurrentVersion\Run\ovegodun: "" C:\ProgramData\(random name).exe ""



```
dwm.exe (728)                              Desktop Window Manager           C:\Windows\system32\dwm.exe
Explorer.EXE (3044)                        Windows Explorer                 C:\Windows\Explorer.EXE
  Procmon.exe (2608)                       Process Monitor                  E:\tool analysis\ProcessMonitor\
    Procmon.exe (928)                      Process Monitor                  E:\tool analysis\ProcessMonitor\
  DOMSTOLSPROCESS_17924.exe (3744)                                          E:\cryptowall zip 24-03-2016\303
    DOMSTOLSPROCESS_17924.exe (276)                                         E:\cryptowall zip 24-03-2016\303
      DOMSTOLSPROCESS_17924.exe (2008)                                      E:\cryptowall zip 24-03-2016\303
      explorer.exe (3248)                  Windows Explorer                 C:\Windows\explorer.exe
        vssadmin.exe (2712)                Command Line Interface for Mic...C:\Windows\SYSTEM32\vssadn
          conhost.exe (2500)               Console Window Host              C:\Windows\system32\conhost.e

Description:
```

Figure 8 : Analysis using Process Monitor

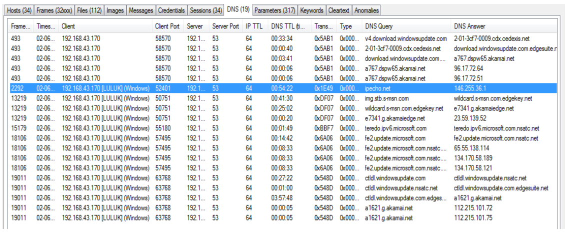
Figure 8 illustrates the environment when the ransomware is run. There are several processes that are running, such as explorer.exe, vssadmin.exe and conhost.exe.

Vssadmin.exe used by the administrator to set the Shadow Volume Copies on a computer. The shadow volume copy allows Windows to take automatic or manual backups, or snapshots, of the current state of the files on a particular volume (drive letter).

The conhost.exe is the new host process for console windows. Previously those were handled by csrss.exe which is the "Client Server Runtime Process", a process running with system-level privileges.

The result showed that the ransomware has prepared some modules such as ntdll.dll, kernel32.dll, kernelbase.dll, user32.dll, shell32.dll, advapi32.dll, gdi32, comctl32.dll, ole32.dll, version.dll, msvcrt.dll, combase.dll, shlwapi.dll, sechost.dll, rpcrt4.dll, imm32.dll, msctf.dll, cryptbase.dll, bcryptprimitives.dll, uxtheme.dll, shfolder.dll, SHCore.dll, oleaut32.dll, clbcatq.dll, propsys.dll, profapi.dll, setupapi.dll, cfgmgr32.dll, devobj.dll, system.dll, camp.dll.

Next process is to detect DNS activity, at this stage Networkminer and ApateDNS are used for analysis. Networkminer has analyzed the DNS addresses that will be used by ransomware to communicate.

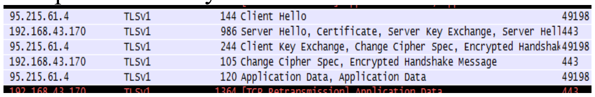


Frame...	Times...	Client	Client Port	Server	Server Port	IP TTL	DNS TTL (ti...	Trans...	Type	DNS Query	DNS Answer
493	02-06...	192.168.43.170	58570	192.1...	53	64	00:33:34	0x5AB1	0x000...	v4.download.windowsupdate.com	2-01-3cf7-0009.cdx.cedexis.net
493	02-06...	192.168.43.170	58570	192.1...	53	64	00:00:40	0x5AB1	0x000...	2-01-3cf7-0009.cdx.cedexis.net	download.windowsupdate.com.edgesuite.n
493	02-06...	192.168.43.170	58570	192.1...	53	64	00:03:41	0x5AB1	0x000...	download.windowsupdate.com....	a767.dspw65.akamai.net
493	02-06...	192.168.43.170	58570	192.1...	53	64	00:00:06	0x5AB1	0x000...	a767.dspw65.akamai.net	96.17.72.64
493	02-06...	192.168.43.170	58570	192.1...	53	64	00:00:06	0x5AB1	0x000...	a767.dspw65.akamai.net	96.17.72.51
2292	02-06...	192.168.43.170 [LULUK] (Windows)	52401	192.1...	53	64	00:54:22	0x1E49	0x000...	ipecho.net	146.255.36.1
13219	02-06...	192.168.43.170 [LULUK] (Windows)	50751	192.1...	53	64	00:41:30	0xDF07	0x000...	img.stb.s-msn.com	wildcard.s-msn.com.edgekey.net
13219	02-06...	192.168.43.170 [LULUK] (Windows)	50751	192.1...	53	64	00:25:02	0xDF07	0x000...	wildcard.s-msn.com.edgekey.net	e7341.g.akamaiedge.net
13219	02-06...	192.168.43.170 [LULUK] (Windows)	50751	192.1...	53	64	00:00:20	0xDF07	0x000...	e7341.g.akamaiedge.net	23.59.139.52
15179	02-06...	192.168.43.170 [LULUK] (Windows)	55180	192.1...	53	64	00:01:49	0xBBF7	0x000...	teredo.ipv6.microsoft.com	teredo.ipv6.microsoft.com.nsatc.net
18106	02-06...	192.168.43.170 [LULUK] (Windows)	57495	192.1...	53	64	00:14:42	0x6A06	0x000...	fe2.update.microsoft.com	fe2.update.microsoft.com.nsatc.net
18106	02-06...	192.168.43.170 [LULUK] (Windows)	57495	192.1...	53	64	00:08:33	0x6A06	0x000...	fe2.update.microsoft.com.nsatc....	65.55.138.114
18106	02-06...	192.168.43.170 [LULUK] (Windows)	57495	192.1...	53	64	00:08:33	0x6A06	0x000...	fe2.update.microsoft.com.nsatc....	134.170.58.189
18106	02-06...	192.168.43.170 [LULUK] (Windows)	57495	192.1...	53	64	00:08:33	0x6A06	0x000...	fe2.update.microsoft.com.nsatc....	134.170.58.121
19011	02-06...	192.168.43.170 [LULUK] (Windows)	63768	192.1...	53	64	00:27:22	0x548D	0x000...	ctldl.windowsupdate.com	ctldl.windowsupdate.nsatc.net
19011	02-06...	192.168.43.170 [LULUK] (Windows)	63768	192.1...	53	64	00:01:00	0x548D	0x000...	ctldl.windowsupdate.nsatc.net	ctldl.windowsupdate.com.edgesuite.net
19011	02-06...	192.168.43.170 [LULUK] (Windows)	63768	192.1...	53	64	03:57:48	0x548D	0x000...	ctldl.windowsupdate.com.edges...	a1621.g.akamai.net
19011	02-06...	192.168.43.170 [LULUK] (Windows)	63768	192.1...	53	64	00:00:05	0x548D	0x000...	a1621.g.akamai.net	112.215.101.72
19011	02-06...	192.168.43.170 [LULUK] (Windows)	63768	192.1...	53	64	00:00:05	0x548D	0x000...	a1621.g.akamai.net	112.215.101.75

Figure 9 : Analysis using networkminer

Figure 9 shows some of the domains that were contacted. The result showed the ipecho.net domain, the domain which is used to find out the ip public from computer victims. Other information has been obtained and showed that there are several domains frequently contacted such as redtable.biz.

Further analysis by Wireshark, showed the data packets sent by ransomware.



```
95.215.61.4        TLSv1          144 Client Hello                                               49198
192.168.43.170     TLSv1          986 Server Hello, Certificate, Server Key Exchange, Server Hell443
95.215.61.4        TLSv1          244 Client Key Exchange, Change Cipher Spec, Encrypted Handshak49198
192.168.43.170     TLSv1          105 Change Cipher Spec, Encrypted Handshake Message            443
95.215.61.4        TLSv1          120 Application Data, Application Data                         49198
192.168.43.170     TLSv1         1364 [TCP Retransmission] Application Data                      443
```

Figure 10 : Analysis using wireshark

From the results of the analysis done by wireshark, there are several result obtained such as:

1. Ransomware communicate with ipecho.net to get ip public of victim.
2. Figure 10 showed communication with multiple servers with ssl Protocol, to secure the communication between a victim's computer and the server.

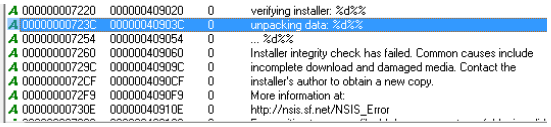
In third section, the ransomware analysis was conducted by Static Code and the tools which are utilized can be seen as follow:

a. Dependency Walker

b. BinText
c. OllyDbg

The analysis has been done by dependency walker, and the result can be seen as follow:

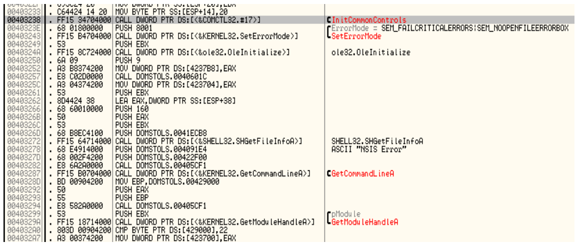
1. Ransomware was able to duplicate or replace the file with another file.
2. The Ransomware was able to count or calculate the files that will be used as the target of infection.
3. The Ransomware has capability related with Anti-Detection/Stealthyness.
4. The Ransomware run CMD and perform a specific command.
5. It is able to collect information inside victim's computer.
6. Other capability on manipulating.
7. The ransomware was able to add/remove/change data in the registry.
8. The Ransomware made itself run automatically when the windows in turn on.



```
A 000000007220   000000409020   0   verifying installer: %d%%
A 00000000723C   00000040903C   0   unpacking data: %d%%
A 000000007254   000000409054   0   ... %d%%
A 000000007260   000000409060   0   Installer integrity check has failed. Common causes include
A 00000000729C   00000040909C   0   incomplete download and damaged media. Contact the
A 0000000072CF   0000004090CF   0   installer's author to obtain a new copy.
A 0000000072F9   0000004090F9   0   More information at:
A 00000000730E   00000040910E   0   http://nsis.sf.net/NSIS_Error
```

Figure 11 : Analysis using Bintext

Figure 11 showed the results by Bintext that the Ransomware was doing unpacking data. Packed is used by ransomware to protect itself from malware analyst software.

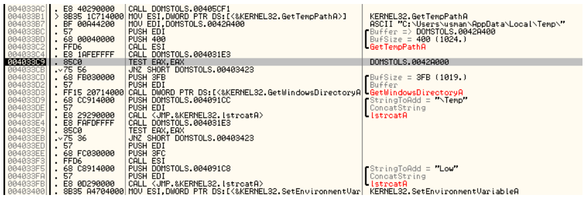


```
00403233 . C64424 14 20   MOV BYTE PTR SS:[ESP+14],20
00403238 . FF15 34704000  CALL DWORD PTR DS:[<&COMCTL32.#17>]          InitCommonControls
0040323E . 68 01800000    PUSH 8001                                     ErrorMode = SEM_FAILCRITICALERRORS|SEM_NOOPENFILEERRORBOX
00403243 . FF15 B4704000  CALL DWORD PTR DS:[<&KERNEL32.SetErrorMode>]  SetErrorMode
00403249 . 53             PUSH EBX
0040324A . FF15 8C724000  CALL DWORD PTR DS:[<&ole32.OleInitialize>]    ole32.OleInitialize
00403250 . 6A 09          PUSH 9
00403252 . A3 B8374200    MOV DWORD PTR DS:[4237B8],EAX
00403257 . E8 C02D0000    CALL DOMSTOLS.0040601C
0040325C . A3 04374200    MOV DWORD PTR DS:[423704],EAX
00403261 . 53             PUSH EBX
00403262 . 8D4424 38      LEA EAX,DWORD PTR SS:[ESP+38]
00403266 . 68 60010000    PUSH 160
0040326B . 50             PUSH EAX
0040326C . 53             PUSH EBX
0040326D . 68 B8EC4100    PUSH DOMSTOLS.0041ECB8
00403272 . FF15 64714000  CALL DWORD PTR DS:[<&SHELL32.SHGetFileInfoA>] SHELL32.SHGetFileInfoA
00403278 . 68 E4914000    PUSH DOMSTOLS.004091E4                        ASCII "NSIS Error"
0040327D . 68 002F4200    PUSH DOMSTOLS.00422F00
00403282 . E8 6A2A0000    CALL DOMSTOLS.00405CF1
00403287 . FF15 B0704000  CALL DWORD PTR DS:[<&KERNEL32.GetCommandLineA>] GetCommandLineA
0040328D . BD 00904200    MOV EBP,DOMSTOLS.00429000
00403292 . 50             PUSH EAX
00403293 . 55             PUSH EBP
00403294 . E8 582A0000    CALL DOMSTOLS.00405CF1
00403299 . 53             PUSH EBX                                      pModule
0040329A . FF15 18714000  CALL DWORD PTR DS:[<&KERNEL32.GetModuleHandleA>] GetModuleHandleA
004032A0 . 803D 00904200  CMP BYTE PTR DS:[429000],22
004032A7 . A3 00374200    MOV DWORD PTR DS:[423700],EAX
```

Figure 12 : Analysis using Ollydbg

Figure 12 showed that the ransomware register itself with the help of comctl32.dll. Next, the ransomware manipulate the system to not showing the error dialog box when an error occurs.

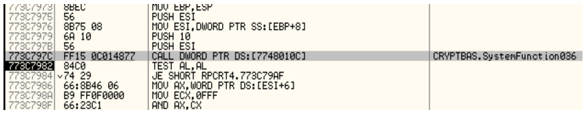
.



```
004033AC . E8 40290000    CALL DOMSTOLS.00405CF1
004033B1 > 8B35 1C714000  MOV ESI,DWORD PTR DS:[<&KERNEL32.GetTempPathA>]  KERNEL32.GetTempPathA
004033B7 . BF 00A44200    MOV EDI,DOMSTOLS.0042A400                       ASCII "C:\Users\usman\AppData\Local\Temp\"
004033BC . 57             PUSH EDI                                        Buffer => DOMSTOLS.0042A400
004033BD . 68 00040000    PUSH 400                                        BufSize = 400 (1024.)
004033C2 . FFD6           CALL ESI                                        GetTempPathA
004033C4 . E8 1AFEFFFF    CALL DOMSTOLS.004031E3
004033C9 . 85C0           TEST EAX,EAX                                    DOMSTOLS.0042A000
004033CB .v75 56          JNZ SHORT DOMSTOLS.00403423
004033CD . 68 FB030000    PUSH 3FB                                        BufSize = 3FB (1019.)
004033D2 . 57             PUSH EDI                                        Buffer
004033D3 . FF15 20714000  CALL DWORD PTR DS:[<&KERNEL32.GetWindowsDirectoryA  GetWindowsDirectoryA
004033D9 . 68 CC914000    PUSH DOMSTOLS.004091CC                          StringToAdd = "\Temp"
004033DE . 57             PUSH EDI                                        ConcatString
004033DF . E8 29290000    CALL <JMP.&KERNEL32.lstrcatA>                   lstrcatA
004033E4 . E8 FAFDFFFF    CALL DOMSTOLS.004031E3
004033E9 . 85C0           TEST EAX,EAX
004033EB .v75 36          JNZ SHORT DOMSTOLS.00403423
004033ED . 57             PUSH EDI
004033EE . 68 FC030000    PUSH 3FC
004033F3 . FFD6           CALL ESI
004033F5 . 68 C8914000    PUSH DOMSTOLS.004091C8                          StringToAdd = "Low"
004033FA . 57             PUSH EDI                                        ConcatString
004033FB . E8 0D290000    CALL <JMP.&KERNEL32.lstrcatA>                   lstrcatA
00403400 . 8B35 A4704000  MOV ESI,DWORD PTR DS:[<&KERNEL32.SetEnvironmentVar  KERNEL32.SetEnvironmentVariableA
```

Figure 13 : Analyzing use Ollydbg

Figure 13 showed that the ransomware created the space for itself on the drive: C:%appdata\local\Temp%



```
773C7973  8BEC            MOV EBP,ESP
773C7975  56              PUSH ESI
773C7976  8B75 08         MOV ESI,DWORD PTR SS:[EBP+8]
773C7979  6A 10           PUSH 10
773C797B  56              PUSH ESI
773C797C  FF15 0C014877   CALL DWORD PTR DS:[7748010C]   CRYPTBAS.SystemFunction036
773C7982  84C0            TEST AL,AL
773C7984 v74 29           JE SHORT RPCRT4.773C79AF
773C7986  66:8B46 06      MOV AX,WORD PTR DS:[ESI+6]
773C798A  B9 FF0F0000     MOV ECX,0FFF
773C798F  66:23C1         AND AX,CX
```

Figure 14 : Analyzing use Ollydbg

Figure 14 showed that the ransomware are calling the module cryptbas.dll and bcryptprimitives.dll. The modules handles the encryption process of files in computer victim.

Until this part, the Ransomware is analyzed based on Surface, Runtime and Static Code. There are several result can be seen as follows: Ransomware cryptolocker is trojan type malware and it was created on Tuesday, October 07, 2014. The targets are all versions of Windows OS, including Windows XP, Windows Vista, Windows 7, and Windows 8.

On the early step, ransomware will unpack itself, then do the communication with the server to get the RSA public key. Ransomware scan computer to find out the data and encrypt it by using AES encryption so that the file can not be opened.

AES key will be encrypted by the RSA public key. Once the infected files are encrypted, the computer displays the message on Notepad or HTML which contains the instructions on how to get the Cryptolocker Decryption Service. The ransomware forces victims to pay a ransom in order to get the key for decrypting the infected files. Victims must pay through unique Bitcoins address.

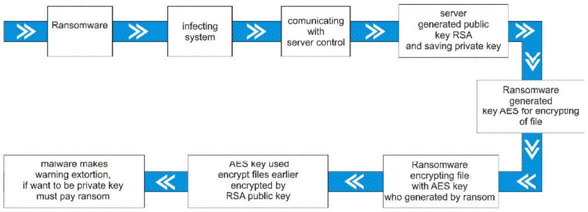

Ransomware
infecting system
comunicating with server control
server generated public key RSA and saving private key
Ransomware generated key AES for encrypting of file
Ransomware encrypting file with AES key who generated by ransom
AES key used encrypt files earlier encrypted by RSA public key
malware makes warning extortion, if want to be private key must pay ransom

Figure 15 : Process of Ransomware

The results from analysis showed that the ransomware duplicates it self into the system folder on following computer address: C:\user\user name\appdata\local\Temp\ <random> \ <random> .exe

To stop infection of cryptolocker, the additional setting on security policies are needed as can be seen in Table 4. By configuring the security policies in the operating sytem, it is expected to prevent the cryptolocker attack.

Table 4. Configuration of local security policies

No	Name	Type	Security Level	Description
1	%AppData%*.exe	Path	Disallowed	Block the existence of the executable file in Appdata
2	%AppData**.exe	Path	Disallowed	Block the existence of the executable files from Appdata
3	%Temp%\Rar**.exe	Path	Disalowed	Block executable file from winrar
4	%Temp%\7z**.exe	Path	Disalowed	Block executable file from 7zip
5	%Temp%*.zip*.exe	Path	Disalowed	Block any executable file inside compressed file zip

7. CONCLUSION

In order to analyze the ransomware, the surface method tried to identify the program whether it is classified as ransomware or not. Next step, it detects obfuscated/package that protects the ransomware and revealed the creation time of application. On the other hand, the Runtime method will provides the information only when the malware is executed. The runtime method allows user to analyze several activities created by ransomware such as the change on registry, monitoring the activities on the system file, monitoring the processes and threads that occured and data communication performed by malware with server. In addition, by utilizing static code method, it provides the information that had not been found by other previous methods such as the ransomware were able to hide from surveillance of computer security system and able to turn off the firewall and antivirus.

Based on this study, the integration of the three methods such as surface, runtime and static code was able to give more detail the characteristics of ransomware, thus the prevention of ransomware attack can be presented. Other limitation is that the the analysis process takes longer time. In the future, minimizing process time for analysis is needed and more detailed results are expected.

REFERENCES

[1] Suryadhi, A. (2012, 12 17). The IT Security issues of Cyber Espionage Until 2013 ' The Insane ' Madware. Retrieved 09 14, 2014, from detikinet: http://inet.detik.com/read/2012/12/17/131135/2120417/323/3/spionase-cyber-hingga-si-gila-madware

[2] Intelligence, D. S. (2014, 8 27). CryptoWall Ransomware. Retrieved 12 28, 2016, from SecureWork: http://www.secureworks.com/cyber-threat-intelligence/threats/cryptowall-ransomware/

[3] Distler, D. (2007). Malware Analysis: An Introduction. SANS Institute InfoSec Reading Room (pp. 18-19). SANS Institute.

[4] Flores, R. (2012, 06 01). Retrieved 05 06, 2015, from https://www.exploit-db.com: https://www.exploit-db.com/docs/18387.pdf

[5] Konstantinou, E. (2008). Metamorphic Virus: Analysis and Detection. Egham, Surrey TW20 0EX, England: Department of Mathematics, Royal Holloway, University of London.

[6] Malhotra, B. (2016). A Survey on Various Malware Detection Techniques on Mobile Platform. IJCA Online, 15-20.

[7] Nugroho, H. A., & Prayudi, Y. (2014). The use of the technique of reverse engineering in malware analysis for identification of malware attacks. National Conference information system 2014. Macassar: Dipanegara STIMIK

[8] Yusirwan, S., Prayudi, Y., & Riadi, I. (2015). Implementation of Malware Analysis using Static and Dynamic Analysis Method. International Journal of Computer Applications , 11-15.

[9] Perdhana, M. R. (2011). Harmless Hacking Malware Analysis dan Vulnerability Development. Yogyakarta: Graha Ilmu.

[10] Senthil Arasu & Barathi. (2014, January). Retrieved 06 06, 2015, from http://ijarcsms.com/docs/paper/volume2/issue1/V2I1-0018.pdf

[11] Michael Sikorski & Andrew Honig. (2012). Practical Malware Analysis The Hands-On Guide to Dissecting Malicious Software. No Starch Press.

[12] Uppal, D., Mehra, V., & Verma, V. 2014. Basic survey on Malware Analysis, Tools and Techniques. International Journal on Computational Sciences &Applications (IJCSA) Vol.4, No.1, February 2014.

[13] Pranoto, E. D. (2007). The ART of Debugging. Yogyakarta: Andi Publisher.