Global Information Assurance Certification Paper

Indicators of Compromise
TeslaCrypt Malware

GIAC Certified Intrusion Analyst (GCIA) Gold Certification

Author: Kevin Kelly,
Email addresses: kjkellyus@yahoo.com
Advisor: Johannes Ullrich
Accepted: 02/16/2017

Template Version September 2014

Abstract

Malware has become a growing concern in a society of interconnected devices and real-time communications. This paper will show how to analyze live ransomware malware samples, how malware processes locally, over time and within the network. Analyzing live ransomware gives a unique three-dimensional perspective, visually locating crucial signatures and behaviors efficiently. In lieu of reverse engineering or parsing the malware executable's infrastructure, live analysis provides a simpler method to root out indicators. Ransomware touches just about every file and many of the registry keys. Analysis can be done, but it needs to be focused. The analysis of malware capabilities from different datasets, including process monitoring, flow data, registry key changes, and network traffic will yield indicators of compromise. These indicators will be collected using various open source tools such as Sysinternals suite, Fiddler, Wireshark, and Snort, to name a few. Malware indicators of compromise will be collected to produce defensive countermeasures against unwanted advanced adversary activity on a network. A virtual appliance platform with simulated production Windows 8 O/S will be created, infected and processed to collect indicators to be used to secure enterprise systems. Different tools will leverage datasets to gather indicators, view malware on multiple layers, contain compromised hosts and prevent future infections.

1. Introduction

Malware analysis performed on local devices requires carefully handling, monitoring, and analysis with proper tools to gather intelligence. The six primary phases of incident handling will be used through the process, including preparation, identification, containment, eradication, recovery, and lessons learned (Skoudis, Strand 2015). A concentration on the first three phases will provide the necessary structure to handle malware incidents. A portable system consisting of virtual appliances will be created to analyze indicators to isolate and contain an incident.

The preparation stage will introduce the process of building a testing environment using open source tools and duplicating the Windows production environment using virtual appliances. Open source tools provide an efficient and cost-effective way to gather information to determine indicators of compromise that may prevent malware from causing damage or compromising a network. The identification process will enable a first responder to quickly identify, contain and eradicate the malware infection. The indicators are gathered from different layers including system and packet level. Indicators can be used to detect malicious activity and files through endpoint malware detection software. When the endpoint software finds a compromised host, steps can be taken to isolate and contain infected devices by port blocking, shutting down services and segregating the infection.

Live ransomware samples will find indicators of compromise (IOC). One ransomware variant, the TeslaCrypt Trojan, is from a global network enterprise infection. A typical TeslaCrypt Trojan incident originates from a help desk call with a user not being able to access files. Some preliminary work is required to locate the original malware sample. A forensic analysis of the host and proxy log analysis will identify malicious activity providing a copy of the executable for testing. Internet browsing streams should be examined through a correlation of proxy log activity to get a general idea of the extent of a compromise and to locate the malware during forensic analysis. The malware sample is then viewed in a controlled environment with various software

Kevin Kelly, kjkellyus@yahoo.com

applications recording events to datasets that will be the basis of finding indicators of compromise to harden a network, protecting it against present and future attacks.

The malware analyst can identify what the ransomware is exploiting and gather indicators of compromise to contain the infection, preventing further contamination. The recovery stage will cover the options companies may have at their disposal to remediate the incident. Ransomware malware is unique in that some businesses may be forced to either shut down without access to their data, or pay the ransom, an ethical and legal issue. Paying off the ransomware has many issues for a business to carefully contemplate. Paying ransom contributes to support of criminal and terrorist activity. The files may not be decrypted or the decryption key may decrypt only some files after paying the ransom, requiring additional payment. The essential point is that the host was compromised with a strong possibility the system was infected with other types of malware.

The final stage will be to review the incident lead by the head incident handler with primary stakeholders; incident handlers will reach a consensus on the business impact to prevent and reduce future incidents. In an enterprise network environment, this will include how to use the indicators of compromise across the endpoints to strengthen any security gaps.

A basic understanding and knowledge of installing Windows, Linux operating systems and software, as well as handling and configuration of hardware, are required to understand concepts put forth here. The primary goal is to provide additional information, retrieve other indicators, and learn what the malware is attempting to accomplish, in order to prevent and contain the infection as quickly as possible. Indicators to look for include IP addresses, file naming conventions, file signatures, services started, registry key changes and anything that can identify a malware campaign. Dropping malware in a sandbox environment may give many of the indicators, depicted and provided via basic screen shots. Analyzing in a real-time environment with many tools collecting and viewing events at different layers provides a more informative, deeper understanding of malicious events, including data exfiltration, denial of service, and dropping of additional malware. Indicators from a sandbox compared to live analysis

is like the difference between two and three dimensional views. Malware analysis from different perspectives in a live analysis will more efficiently secure a network, isolate and contain the infection, adding more value to the investigation.

2. Building Virtual Appliances to Monitor Malware

There are two ways to build out a representative host using production standards to analyze malware. The first way is to create a 'bare-metal box,' meaning an actual physical workstation. The second is to use a virtual appliance. Both ways require preconfigured tools installed to analyze malware. A bare-metal box may be the only solution if the malware is VM- (Virtual Machine) aware. VM-aware malware may limit its effectiveness, preventing attacks on many enterprise networks where virtualization is the norm. There are benefits of using a bare-metal box for analysis, including the creation of a duplicate copy of the original system, hardware which can mirror actual production systems, not giving away a business's configurations, certificates, or other security keys. Drawbacks include the cost of the computer hardware, limited control of physical hardware, and extra space needed for each device.

The second option is to create VMs with a similar operating system and software applications found on production hosts. A VM appliance gives plenty of configuration options, including a virtual network which connects multiple hosts, using different operating systems, various software versions, easy adjustments to hardware simulation of memory, hard drive space and the number of CPU processors. One option is to reduce the memory and CPUs being used to slow down the malware processing. Indicators may be found more readily when the process occurs in a few seconds as opposed to a fraction of a second.

Both a bare-metal box or VM appliance can have issues caused by prepackaged software installation and configurations from third-party manufacturers. Some essential tips to prevent false positive results is to start with the basics: wiping the hard drive and installing a fresh copy of the operating system acquired directly from the operating system vendor. Almost every computer bought directly from a manufacturer has a

preinstalled operating system containing many unnecessary programs that reduce the cost of the computer. The added third party software resides in the computer's memory when booted up and similar to adware, alters configurations. Reducing the interference of third party programs frees hardware resources from being tied up. As a corollary to this measure, the administrator should not restore from the manufacturers' restoration or recovery disks. Creating a system from scratch will take a bit more time in the beginning, but allows more control over the environment to reduce false positive results and noise during the time of analysis.

The preferred method to build a bare-metal box with the necessary tools is to make a duplicate copy of the physical hard drive, preserving a master copy. Performing tests on a system built similarly to a production host will yield results representative of the malicious behavior occurring in production. It will be difficult to keep the system up to date with updates and patches, but if the software version is at or behind the production build, results of an infection will project more reliable details of the incident.

3. Flexibility of Using Virtual Appliances with Ransomware

The flexibility of a virtual environment gives the best method for examining ransomware malware. In this case, VMware Workstation 12, will be used to run the VM appliances. Monitoring software can be installed within the virtual machine or on the workstation running the VMware Workstation application. Ransomware often encrypts files on simulated virtual drives, encrypting all files indiscriminately, including the monitoring software data as well as connected drives. Recording devices such as Bandicam or Windows Snipping tool outside of the VMware environment can monitor and preserve the data for a later review and analysis quicker than retrieving through computer forensic processing. The snapshot feature of VMware Workstation is another way to gather data as the malware progresses at different stages.

Kevin Kelly, kjkellyus@yahoo.com

4. Simulating a Production Host with Monitoring Software

Virtual appliances add a whole new dimension for analyzing malware. Virtualization offers a way to record applications from an outside source by placing recording software on the same operating system used to host the VMware software, giving an outside looking in point of view. The VMware snapshot feature allows for the creation of a stopping point to easily revert back to the state of the VM appliance when the snapshot was taken. A malware investiagtor can create different configurations of a VM appliances by saving each one separately. The many different configured appliances may house a unique varienty of monitoring tools for testing or analyzing within the context of an externally Internet connected or self contained internal virtual network. Upgrade and update baseline VM appliances to mimic the production environment, using different generations of software in various VM appliances. Exploit-kits may run through a list of vulnerabilities using the first exploit detected, therefore a VM appliance that is not an accurate representation of a production device may provide a set of false positive indicators of compromise.

5. Tools

A collection of applications to monitor data will be added to the simulated production host build, accumulating evidentiary artifacts caused by the malware infection. The Sysinternals suite of tools is an essential requirement and may be used exclusively in some evaluations of malware. Sysinternals' Process Monitor (Procmon.exe) gathers all instructions to the CPU, but note that this collects hundreds of thousands of instructions within a minute or two, to be used only for brief periods of time. Procmon is turned on before the infection and shut off after the initial fireworks have subsided. There are many other Sysinternals tools, including autoruns.exe, autorunsc.exe, procexp.exe, tcpview.exe. Sysinternals' Process Explorer and TCP View offer additional monitoring of real-time events to trace process trees and processes

connected to ports respectively, but as a snapshot in time. The point here is that some tools allow for scrolling and accumulating data, whereas others need data samples saved periodically.

6. Examining Ransomware Malware Case Study

Two samples of malware will be studied. The two samples of the TeslaCrypt ransomware are MD5 sum f3b12a197d732cda29d6d9e698ea58bf (RansomWareSample.exe) and 5df8b61f8355fa08eb90d6d2837dba0e (RansomWareSampleM.exe), both downloaded from https://malwr[.]com website. Copies of the Malware for this paper are downloaded from the public website to ensure no company information is publicly exposed, directing no malicious activity at a targeted organization. Sending malware files found on an enterprise system to vendors requires additional scrutiny not provided here. Both samples are variants of Tesla Crypt and encrypt files on the C: drive. The RansomWareSampleM.exe sample has an additional feature of encrypting files on all mapped drives and mapped network shares folders. The original malicious files were also on other hosts under various names: 95fb.tmp, hhptdwp.exe, kinkhw.exe, 54B9.tmp. All infected systems were Windows based with the initial malware operating from the \AppData\Local\Temp\ folder. For this paper, the two similar ransomware Trojan samples will be used to gather knowledge of what the malware did as well as the scope of the files touched.

The reason to explore two variants of the same family of malware is to discover the indicators and scope of the infection using the virtual appliance or bare-metal box, representing the host. The first malware, RansomWareSample.exe, encrypts files on the local drive, whereas RansomWareSampleM.exe parses through the mapped drives to locate files. Malware parsing mapped drives and network share drives makes it harder to save datasets from tools on an infected host before the data is encrypted.

There were two methods used to execute the malware, one through a phishing campaign, the other through an exploit operated through the Angler Exploit Kit. Clicking a phishing link in an email triggered execution of the malicious Trojan file. The other

case clicked on a link, leading to a landing page with the Angler Exploit Kit, determined by viewing the user's activity at the time of the infection. The malware downloaded during a time the user was performing research on the Internet. An investigator would forensically pull and examine the cache files to determine the root-cause. A cached file contained evidence of a hidden iframe, with negative number assigned to a position variable making the download and execution of the ransomware invisible on the user's monitor. Open source intelligence, in this case, Palo Alto Research Center, provides an excellent forum to learn and understand the Angler Exploit Kit (Duncan, 2016). Correlation of events by analyzing proxy log data provides download streams of activity, allowing a quick, forensic examination to retrieve ransomware malware files based on timestamps.

Kevin Kelly, kjkellyus@yahoo.com

7. Tools versus Indicators of Compromise

The chart below (Illustration 1) provides a listing of tools and the indicators of compromise (IOC) findings in an analysis of the live execution of malware. The IOCs are in the top row and the tools used are in the first column. An X will signify an indicator of compromise may be determined entirely by that application. P will signify a partial finding needing additional collaboration from other resources and D represents using the IOC to detect malware attacking the system. How to use the tools mentioned to locate an IOC will be explained in detail in following sections.

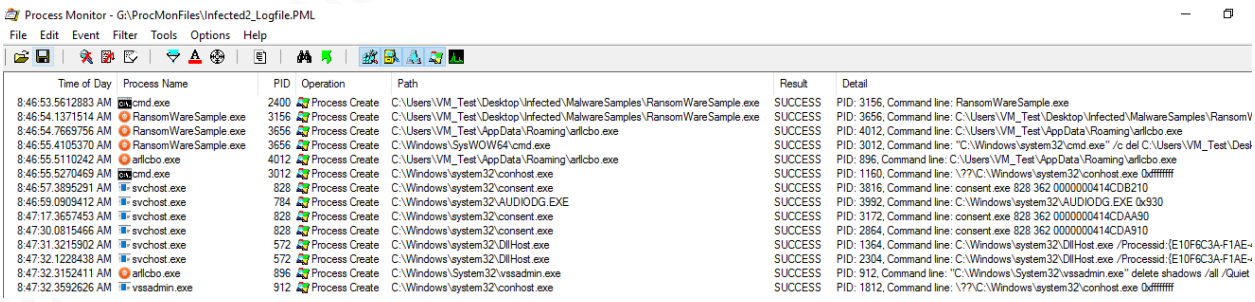
Tool	Randomwar eSample.exe	arllcbo.exe	Log.html Key.dat	HELP_RESTO RE_FILES.txt	7tno4hib47v lep5o.63ghd	State1.php	Encrypted files *.ecc	key	ipinfo.io/ip
Procmon	X	X	X	X					
Autorun autorunsc		X							
Tcpview		P			P				
Fiddler					X	X		X	
Wireshark					X	X		X	X
Procexp	X	X							
MD5deep	P	P							
Pslist		P			P				
Listdlls	X	X							
Forensic Examiner	PD	PD	PD	PD	D	PD	PD	X	
HexEditor			P	P			P	X	
Snort					D	D			

Tools Used to Find Indicators of Compromise
Illustration 1

Kevin Kelly, kjkellyus@yahoo.com

8. Process Monitor

A crucial way to protect a network from a previously known attack is to find indicators of compromise (IOC). Process Monitor, Procmon.exe, from Sysinternals, is a free, advanced monitoring tool for Windows that shows a real-time file system, Registry, and process/thread activity (Russinovich, 2016). In the analysis of ransomware, the malware renames itself then operates out of the \AppData\Roaming\ folder. Process Monitor data is an excellent starting point for finding registry keys, files, and processes touched by the malware on a live system. Process Monitor contains an easy to use filtering capability. The custom filtering can pull different datasets. To gather all environmental variables malware encounters, use the filter **[Details contains %]**. To start finding IOCs, locate any running processes created by the malware, filtering on all processes started since the beginning of capture [**Operation is Process Create**] (Illustration 2).



Process Monitor Filtering on ***Operations is Process Create***

Illustration 2

Illustration 2 provides several IOCs. The malware file RansomWareSample.exe spawns a clone executable file arllbro.exe, then deletes itself. The users \AppData\Roaming\ directory is used to place the cloned secondary malware executable arllcbo.exe. Both files, RansomWareSample.exe and arllcbo.exe, had the same md5 hash values as determined by the tool md5deep64.exe.

A Process Monitor filter [Operation is WriteFile] will locate all files created by the malware, including \AppData\Roaming\key.dat an encryption key, AppData\Roaming\log.html which contains a listing of all encrypted files, and a text file,

HELP_RESTORE_FILES.txt, dropped in directories containing encrypted files. The IOCs here are two distinct files in the \AppData\Roaming\ folder, key.dat and log.html, as well as identical HELP_RESTORE_FILES.txt files dropped in every folder with an encrypted file. The HELP_RESTORE_FILES.txt file contains a text message explaining how to pay ransom to unencrypt the files in case the antivirus stopped the ransomware before the ransomware screen appeared.

The ransomware variant RansomWareSampleM.exe uses vssadmin.exe to. In the illustration above, vssadmin is used by the cloned malware arllcbo.exe to execute the command ***C:\Windows\System32\vssadmin.exe delete shadows /all /Quiet.*** Vssadmin.exe program quietly deletes all shadow volumes and files preventing Windows from recovering any files, in this case. Shadow volume copy service is a technology in Windows that can create a snap shot of files and volumes. Vssadmin is a seldom used administrative tool that may be removed to keep intact the Window shadow volume (Abrams, 2015).

RansomWareSample.exe creates the key.dat file at the inception of execution. Many CreateFile operations are transpiring when filtering out the malware processes by name, RansomWareSample, arllcbo. Later, the file log.html is created and located in the \AppData\Roaming\ folder alongside key.dat. The other fact determined here is most of the files created are files residing on the host system, indicating changes occurring to these files, and in this case, encryption.

The process took approximately 20 minutes to examine and provides a wealth of information in determining what malware is doing on a system. Focusing on processes and files created is a good starting point for locating IOCs generated by the malware. Double-clicking on any item in the list gives further detailed information concerning the event, process, and stack. The events show paths touched, variables used, operations performed, and many other details.

The first infected sample produced ogbofmx.exe, a later version arllcbo.exe, indicating the malware created to control the host is using an evading technique by changing its name. Testing for polymorphic malware is done through a collection of md5 hash values to determine if not only the name changes, but if the executable itself changes. The MD5 hash values of the original malware sample, arllcbo.exe and ogbofmx.exe were

Kevin Kelly, kjkellyus@yahoo.com

identical and not polymorphic. We can filter on the original RansomWareSample.exe and the cloned arllcbo.exe to remove much of the noise in the dataset. Autoruns or autorunsc is used to determine any new processes created by the malware to start automatically.

9. Autorun

Autorun data sets are compared from a pre-infection and post infection state with f3b12a197d732cda29d6d9e698ea58bf ransomware. The two variants of the Autorun application include, autoruns.exe a graphical user interface, applications and autorunsc.exe, a command line version used in this test case. One of the first tests to run is Sysinternals autorunsc.exe within a command prompt which collects the output pre-malware infection and post-malware infection (Illustration 3). Next is to compare any differences using the Microsoft file compare command within a command shell, ***C:\fc.exe pre.infection.txt post.infection.txt***, detecting a new autorun for a file, arllcbo.exe, was created in the users AppData\Roaming\ path.

```
C:\autorunsc.exe > pre.infection.txt          Retrieved prior to infection
C:\autorunsc.exe > post.infection.txt         Retrieved after infection
C:\fc.exe pre.infection.txt post.infection.txt
```

Illustration 3

The registry change, (Illustration 4), allows ransomware executable file arllcbo.exe to execute every time Windows starts. The finding gives us two important IOCs that were consistent on several tests. The malware always resides in the \AppData\Roaming\ folder, and the date from the autorun registry key 12/31/1969 is not normal. The timestamp of the malware is another important indicator of compromise, worthy to note since the name of the malware is random and changes with each execution, while the time stamp remains the same and is unique. A rule can be made to search for computers with AutoRun registry values with the abnormal date located in the user's \AppData\Roaming\ directory.

```
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
  msconfig
    C:\Users\VM_Test\AppData\Roaming\arllcbo.exe
    c:\users\vm_test\appdata\roaming\arllcbo.exe
```

Kevin Kelly, kjkellyus@yahoo.com

12/31/1969 7:00 PM

Illustration 4

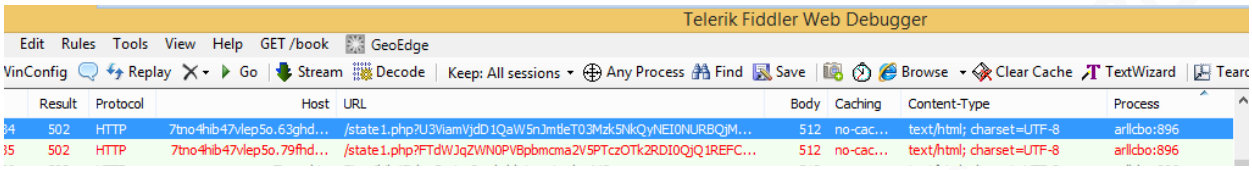
10. Tcpview

Tcpview is a Sysinternals tool, providing "detailed listings of all TCP and UDP endpoints on running on a system, including the local and remote addresses and state of TCP connections" (Russinovich, 2016). Tcpview is a useful tool for finding what process has open port(s) on an active workstation. A drawback of TCPviews is that it refreshes every second by default, allowing the possibility of missing an external communication. TCPview provides process name, process ID, Protocol, data transfer information, and source/destination IP:port. The data is a snapshot in time and does keep a rolling account of TCP/UDP connections. TCPview is frequently recorded during malware execution to get a better understanding of what is transpiring. Fiddler provides a better tool to monitor network sessions, keeping a list of connections over time.

11. Fiddler – Simple Proxy

Fiddler provides a simple proxy to show activity on the network. Fiddler is a portable proxy that can display network flow and more importantly get a quick indication of how much data is coming on or off the network. In this sample, a connection request is made to a file state1.php, an IOC. Another indicator depicts the malware reaching out to a command and control (C2C) server that was not accessible, but attempts were made to connect to host 7tno4hib47vlep5o.63ghdye17[.]com (Illustration 5). Fiddler provides other useful information, including Header, Request Methods, and Cookie data. The Composer tab allows the creation and sending of customized requests by cloning a previous request or composing one from scratch. Creating custom responses is useful in simulating traffic from a previously captured pcap to manipulate the malware's behavior. A more comprehensive analysis of payload data can be conducted using Wireshark.

Kevin Kelly, kjkellyus@yahoo.com

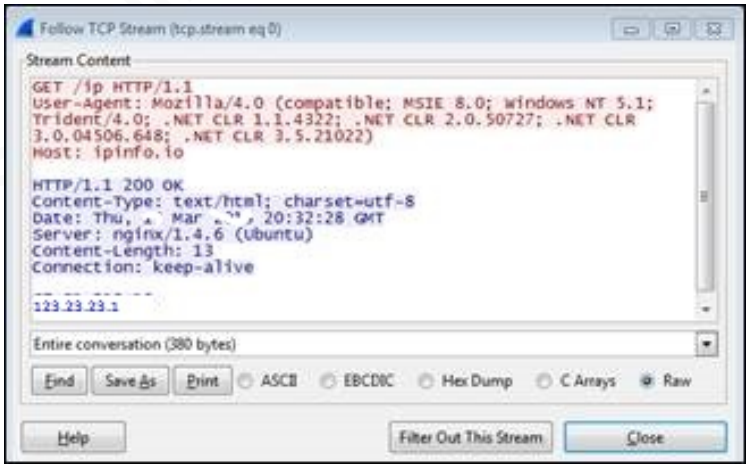


GET Request to state1.php with Base64 Encoded String

Illustration 5

12. Wireshark

Wireshark is a packet capture and analysis tool. Wireshark can be placed on the virtual workstation or networked with other virtual hosts such as an Ubuntu box for Linux-analyzing utilities. RansomWareSampleM.exe malware attains the infected host's IP address then using ipinfo.io, queries for user's ip address (Illustration 6).



Local IP Address 123.23.23.1 Retrieved Using http://ipinfo[.]io/ip

Illustration 6

The results of the information queried from ipinfo.io is the infected host's IP address as text. The infected host IP is passed to the command and control server using a Base64 encoded string passed to the server through a GET request for client side script, state1.php (Illustration 7). Wireshark contains a multitude of tools for analysis and other indicators, gathering details on packet information. The Wireshark commands 'Follow -> TCP Stream' and 'Follow -> UDP Stream' is used to reassemble the packets to

Kevin Kelly, kjkellyus@yahoo.com

determine what information was exfiltrated or infiltrated. Illustration 7 shows a GET request with the decoded Base64 string passing data to a TeslaCrypt command and control server.

GET/state1.php?
Subject=Crypted&key=A546DEB66C3AE1D76E4217F37948FD22D9D97B4
02CC1B4508E5A596C443BB18A&addr=154u36p4v1q9BPcRTBwuay18YG
m52aH1dkF2d&files=275&size=39&version=0.3.2&date=1426192340&OS=7
601&ID=21&subid=0&gate=G0&is_admin=0&is_64=1&ip=123.23.23.1

[Key symbol & concatenates strings, used as a delimiter,
date=1426192340 is converted to 03/12/2015 20:32:20 GMT]

Note: IP *123.23.23.1* represents the infected machines IP address. The IP number and addr field were changed for security purposes.

Illustration 7

Indicators of compromise located in Wireshark include the use of http ://ipinfo[.]io/ip and state1.php GET request. The key may also be used as an indicator, but would need to replicate and examine with several tests. Analysis on image files is the next step in the process.

13. Forensic Analysis

There are many forensic tools available to analyze malware. The original malware and subsequent remnants of information on the host allow a computer to gather malicious files without getting infected or compromising evidence. Connections to a live system to retrieve data will likely lead to getting infected or having files encrypted with ransomware Trojans. A hex editor application performs examination of the files key.dat and log.html on a live system. Malware is better to view on a live system than from a forensic image.

A forensic image of the VM session may be acquired using open source forensic software Guymager or SANS SIFT Workstation. The contents of key.dat viewed as hexadecimal revealed the first 16 bytes were the Bitcoin IP address and the decode key from Illustration 7 seen both in Fiddler and Wireshark analysis. A hex editor determines the log.html file contains a listing of all the encrypted files on the system. IOCs are crucial

for locating malware on file systems that minimally contain over 200,000 + files. Indicators include time/date stamps, file names, file locations, and strings to search on.

IOCs to secure a system pass on to other intrusion detection and prevention systems. The malware is sent to endpoint anti-virus vendors to create a signature for finding and deleting identical files touching an endpoint. The name and location of files log.html and key.dat are good indicators when both are found in the user's \AppData\Roaming\ folder. A directory with HELP_RESTORE_FILES.txt is an indication the malware touched the folder and encrypted files especially files containing a unique .ecc file extension appended to the original file name. The state.php file may reside as an Internet cache file or the string 7tno4hib47vlep5o.63ghdye17[.]com can be searched for, but the best way to use these IOCs is to build rules to use in an intrusion detection and prevention system like Snort.

14. Snort Intrusion Detection Rule

Snort rules allow the intrusion detection administrator to create signatures to alert or stop traffic based on certain criteria. To prevent access to the website 7tno4hib47vlep5o.63ghdye17[.]com a rule can be made to drop any request to a DNS server, preventing the resolution of the IP address (Illustration 8). A separate rule is created for finding the string 'state1.php', but this by itself may lead to false positive results as "state" is a commonly used name. The following rule may be tweaked to perfection within a testing environment then pushed out to production:

drop udp $HOME_NET 53 -> any any (msg:"Tesla Ransomware Website – DNS resolution prevention"; content:"7tno4hib47vlep5o.63ghdye17"; nocase; sid:1123456;)

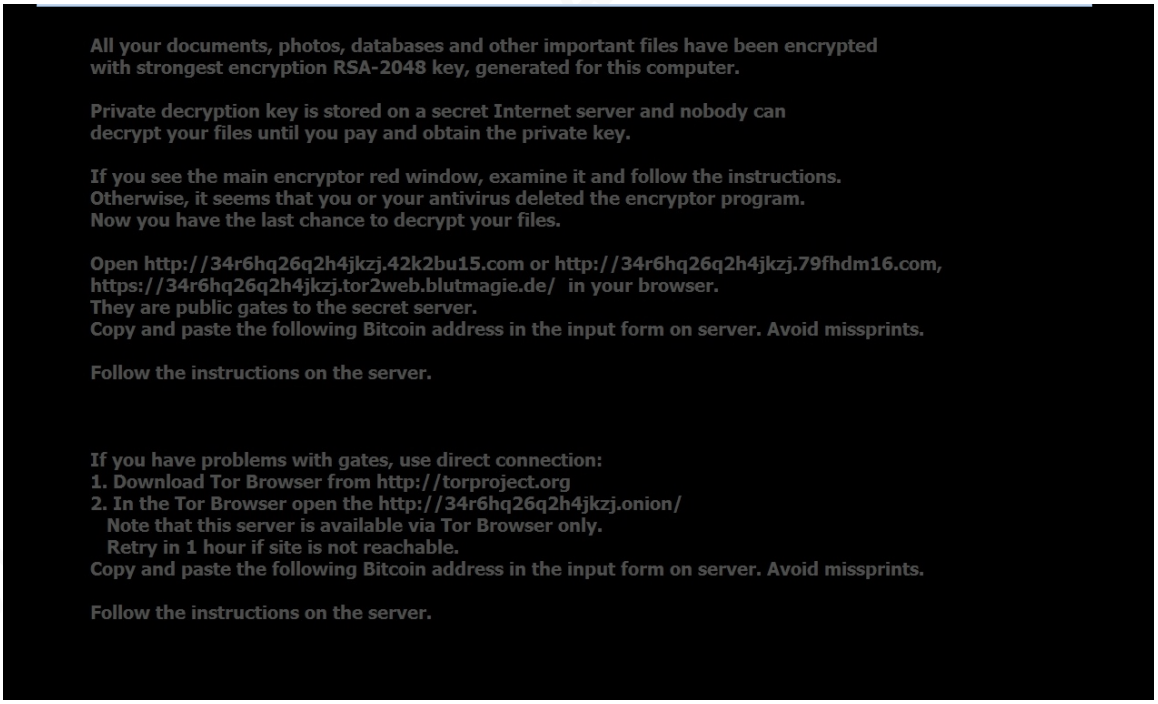
Snort rules to drop dns request to resolve a malicious website

Illustration 8

15. Understanding Ransomware

Kevin Kelly, kjkellyus@yahoo.com

Gathering indicators of compromise and researching online may provide additional insight into the intention of the malware for follow up. This malware indicated by the ransom screen was a version of TeslaCrypt known to target gaming files. We have seen through the process monitoring analysis that python files were a primary target. The original executable did remove itself by creating the file arllcbo.exe which takes control of the system at the startup of Windows OS, encrypting targeted files. Registry keys' changes can be seen through process monitoring, including a shutdown of Windows safe boot to disable all program breaking mechanisms. The following three screens appear after all the files are encrypted and the ransomware controls the workstation.



Background Screen after Ransomware Controls the Compromised Host

Illustration 9

Kevin Kelly, kjkellyus@yahoo.com

Additional Warning Screen with Ransom Paying Scheme and Countdown Timer

Illustration 10

Decryption Website Asking for 1.5 Bitcoins Ransom

Illustration 11

Gathering of additional indicators today is paramount to a successful cyber investigation. To verify and gather additional information from open source information through an Internet search is an often overlooked tool. It may be prudent, in most cases,

Kevin Kelly, kjkellyus@yahoo.com

to find similarities through online search engines. An investigator must also be aware the searching may also tip off hackers that the malware was successful. In this case, the investigator should use his or her judgment. Additional malicious file hash values, websites, or running process, may have been found from similar malware strains to provide more indicators. Even after the fact, open source information may give a better insight in determining the root-cause or finding information on sites taken down. Bromium Lab also reported a ransomware malware incident in an article, 'Achievement Locked: New Crypto-Ransomware Pwns Video Gamers', a week after the original analysis, but provided an answer to the original root-cause through a drive-by download (Kotov, March 2015). Missing pcaps from the original incident, open source information may be the last alternative. The fact of an invisible iframe having a position value with a negative number within an Internet cache files was valuable to the investigation. Bromium Lab also provided an additional Common Vulnerabilities and Exposures (CVE) to review. The article also provided additional insight of mostly targeting online gaming programs. Mitre's CVE archives listed the exploit as, "Unspecified vulnerability in Adobe Flash Player through 13.0.0.262 and 14.x, 15.x, and 16.x through 16.0.0.287 on Windows and OS X and through 11.2.202.438 on Linux allows remote attackers to execute arbitrary code via unknown vectors, as exploited in the wild in January 2015." Basically caused by using an unpatched version of Adobe Flash Player.

16. Ransomware Eliminating threats

The analysis of RansomWareSampleM.exe with a MD5 hash value of 5df8b61f8355fa08eb90d6d2837dba0e had numerous obstacles to overcome. One of the first things performed by this malware is disabling Sysinternals' Process Explorer application, and the Windows command shell. Windows Task Manager application was used to kill the ransomware process. There are many ways to stop or kill processes in a Windows environment if one way is eliminated try to find an alternate way. Killing the ransomware process will allow the saving of datasets from the detection software. The malware also went through all mapped drives, so storing datasets on an external device was difficult. The best process is to record from outside the VMWare all the collection

tools. Narrow the collection to a few key tools, have Window Task Manager running opened on the 'Details' tab will allow the stopping of the Ransomware process for the collection of data.

17. Conclusion

The analysis of malware is an integral part of securing a cyber environment. Any overlooked indicators may lead to reinfection spreading throughout the network. Files shared on network drive are often used as a conduit to spread the infection through the network. Preparing a means of analyzing malware ahead of time can isolate and contain malicious intentions toward a compromised host or network. Building the right platform to analyze does not have to be costly. There are many open source solutions for gathering dataset to analyze malware. Creating a virtual network of tools can leverage viewing from many different layers or dimensions.

A virtual environment is preferred over a bare-metal box from an incident response viewpoint to view malware activity. A USB drive containing all the virtual appliances needed is a worthwhile endeavor to pursue. Creating a virtual system that can be easily added to a shirt pocket or jump bag for use onsite saves time, gathers critical indicators of compromise, preventing further damage by isolating and containing a problem.

Virtual machines hosting various open source tools can be cloned and placed on a thumb drive for ease of portability and then can be transferred to a new examiner when needed. Required devices include simulated production hosts, a vulnerability testing platform, a Linux analyzer, and a forensic examiner. Add additional virtual appliances as needed.

Simulated production host VM appliances require similar specifications to production workstation builds and include additional monitoring software. A virtual platform such as VMware Workstation 12 allows VM sessions to be saved at specific points using the snapshot feature. Monitoring software can be executed one application

at a time or by using multiple applications. If something is not working correctly, reverting to the last snapshot allows a quick change of software or configurations.

Indicators of compromise are a vital component to locating, isolating, containing and preventing damage to a network or workstation. Different tools gather information to provide a deeper understanding of events transpiring in a malware infection. Process Monitor provides a wealth of data that may seem overwhelming, but the filtering system allows the investigator to focus on processes started and used by the malware. Process explorer is usually a good tool to use, but was disabled quickly by the malware. A simple MD5 sum check detected all the malware generated was the same, not polymorphic. A live analysis provides an all-encompassing view at what is happening, as well as how to secure and prevent other machines from compromise.

Looking at the different levels of a host and network level is an essential element of an effective analysis. The order of volatility is critical to gathering datasets. Process Monitor provides an in-depth analysis of files created, commands executed and other information. Autorun is able to find a new malicious executable file created. Fiddler provides a proxy to monitor network flow and session data. Using Fiddler along with Wireshark gives a better understanding of the network conversations and data exchanges.

Utility tools like md5deep64.exe allow investigators to find that the original malware file renames itself and resides in the \AppData\Roaming\ folder. Tcpview gives us information only for a picture in time, a snapshot of data, requiring frequent saves. Tcpview yields a process ID number that needs to collaborate with another program to translate process IDs into process names and locations.

The forensic examination requires indicators to target files for a preservation of evidence, locating additional signs and anomalies on a computer. The last stage is to use the indicators of compromise to protect the enterprise network. Endpoint security vendors take the malware files and other indicators to create signatures, allowing for rapid deployment, rendering malicious activity futile. The last step is writing and deploying rules to intrusion detection and prevention applications. The use of many tools allows a look at different perspectives to quickly find the many indicators of compromise needed to secure an enterprise's network.

Kevin Kelly, kjkellyus@yahoo.com

References

Abrams, Lawrence, (November 2015), Why Everyone Should disable VSSAdmin.exe Now!, Retrieved February 9, 2017 from BleepingComputer Article, https://www.bleepingcomputer.com/news/security/why-everyone-should-disable-vssadmin-exe-now/

Duncan, Brad (June 2016), Understanding Angler Exploit Kit – Part 1: Exploit Kit Fundamentals, Retrieved December 15, 2016 from Palo Alto Networks http://researchcenter.paloaltonetworks.com/2016/06/unit42-understanding-angler-exploit-kit-part-1-exploit-kit-fundamentals/

Kotov, Vadim, (March 2015), *Achievement Locked: New Crypto-Ransomware Pwns Video Gamers*, Retrieved December 29, 2016 from Bromium Labs, https://labs.bromium.com/2015/03/12/achievement-locked-new-crypto-ransomware-pwns-video-gamers/

McCarthy, Ronald (2015), *Bro IDS – Got Your Back*, p34, ADMIN Network and Security Magazine (Feb/Mar 2015 Issue 24).

Novak, Judy, et al (2016), *SANS 503 Intrusion Detection In-Depth*, The SANS Institute.

Russinovich, Mark (2016), *Windows Sysinternals*, Retrieved January 20, 2017, Microsoft, https://technet.microsoft.com/en-us/sysinternals/default

Skoudis, Ed & Strand, John (2015), *SANS 504 Hacker Tools, Techniques, Exploits and Incident Handling*, The SANS Institute.

Turkel, Dan (April 2016), *Victims paid more than $24 million to ransomware criminals in 2015 — and that's just the beginning*, Business Insider, Retrieved December 20, 2016 from http://www.businessinsider.com/doj-and-dhs-ransomware-attacks-government-2016-4.

Kevin Kelly, kjkellyus@yahoo.com