

Ransomware

Project Report

Submitted By Group 8

Prathima Bommannagari

Abhay Arora

Amish Gadhia

Bhavani Gudhollu

Sriteja Puttapaka

Abstract

Ransomware attacks have become global prevalence, with the primary objective of making economic gains through unlawful means. It can result in the loss of sensitive information, regular business operations disruption, and harm an organization's reputation. It encrypts targets' files and displays notifications requesting payment before the data can be unlocked. For a few years, new variations of ransomware have been deployed periodically. As a result, ransomware incidents have become more destructive and impactful in nature and scope. Malicious actors use lateral movement to target critical data and propagate ransomware across entire networks. These actors also increasingly use tactics, such as deleting system backups, that make restoration and recovery more difficult or infeasible for impacted organizations.

Malware accounts for hundreds of millions of dollars of losses annually in ransomware attacks. The ransom demand is usually in the form of virtual currency, bitcoin because it is hard to track. The need of the hour is to gain knowledge about the ransomware threat and accordingly take precautions and measures to prepare for and minimize hazards from ransomware attacks.

This report includes insights into the infection of the crypto-ransomware that encrypts the files and subfolders of a directory on a Linux machine using 128-bit AES symmetric encryption and demands a ransom from the victim in exchange for a decryption key. The report also explores recent advances in the detection and prevention of ransomware and highlights future research challenges and directions. The study's findings are immensely valuable for understanding the effects of ransomware attacks in environments with critical infrastructure and the use of various frameworks or technologies to detect and prevent these attacks.

Keywords: Ransomware, Symmetric Encryption, Decryption, Detection, Mitigation

Introduction

Ransomware is not a new cybersecurity threat in the modern era. However, it has recently caught the attention of the highest levels of government. People's access to medical treatment, gas for their cars, and grocery shopping were all impacted by ransomware. As a result, understanding how these attacks work is critical to prevent them. This report focuses on the application of crypto-ransomware to better understand the attackers' motivations and goals. This also assists us in developing logic to mitigate these attacks.

Firstly, the research contains an understanding of the attackers' minds and ransomware development. First, the encryption technique is chosen for the attack using which the ransomware encrypts all the files and subfolders of the victim's target directory. For the encryption purpose, we use a 128-bit AES symmetric encryption technique. This encryption program is converted into an executable file and infected via a malicious website. Attackers fool victims to download premium software from the malicious website containing the malicious executable. Once the user runs the executable file, all the files on the targeted victim's machine are encrypted and the data is lost. The attacker sends the decryption file with the key only after the ransom is paid.

Secondly, the project focuses on the most crucial part which is building countermeasures against ransomware. The victim machine needs a script or application to monitor all the activity and detect any suspicious files or processes. For that purpose, a python script is employed to monitor the files of the target directory and raise an alert when it detects any encryption activity. Once the monitoring and detection script raises an alert, the victim machine needs to be secured by halting the encryption process.

Finally, a mitigation technique is included in the project to identify the attack-causing process and terminate the process immediately. This helps the victim secure their computer environment from ransomware attacks.

Related works

A New Approach to Detecting Ransomware with Deception, YUN FENG, CHAOGE LIU, BAOXU LIU INSTITUTE OF INFORMATION ENGINEERING.

This paper presented a deception-based and behavior-based method for real-time ransomware detection. In this process, the system produces decoy files for malicious activity detection so that the method causes no loss before ransomware is found. They have conducted a pilot study using Locky, and the results demonstrate the effectiveness of our strategy with little system resource usage and geographical cost. This method would also work for recursive and depth-first file traversal. Before accessing the following file, file-searching threads traverse all files in a directory. It is equivalent to inserting decoy files in each directory using our strategy. As a result, ransomware will scan many decoy files when traversing files. There are few files transferred, which prevents excessive disk use.

File operations are implemented by invoking two Windows APIs (Application Programmer s Interface), FindFirstFile and FindNextFile. (Unicode functions) while FindFirstFileA and FindNextFileA are ANSI functions. FindFirstFile searches a directory for a file or subfolder with a name matching a particular or partial name with wildcards. To continue a file search, FindNextFile is called after FindFirstFile. As a result, we concentrate on any process that calls FindFirstFile or FindNextFile. The hook is a method for determining if a process has been called the two APIs.

The decoy file monitor is used to determine whether a file has been encrypted by comparing the Shannon entropy of the original file and the one updated by an application. There are also numerous successful detection methods, such as file type modification, similarity comparison using sdhash, or a combination of the above methods. If a process exhibits harmful behavior, it will be terminated, and a warning message will be shown on the screen to inform users.

This innovative approach will result in no loss since processes must first operate decoy files and will only be permitted to operate open files after passing the detection. In this way, they can stop the files from encryption.

Multi-layered defense architecture against Ransomware, MANVEER PATYAL SRINIVAS SAMPALLI DALHOUSIE UNIVERSITY, CANADA QIANG YE UNIVERSITY OF PRINCE EDWARD ISLAND,

In this paper, they have studied the Life cycle of the Ransomware attack. Based on this, they have implemented a multi-layer protection architecture built on the ransomware model. At each tier, different strategies are used to serve as a defense or to identify the ransomware capabilities. The multi-layered defensive architecture can deal with all forms of ransomware assaults and would be more successful against such threats than current antivirus methods, which hunt for attack signatures and hence only deal with certain types of ransomware.

In Layer 1, Better policies must be implemented other than just asking for user Permission; policies like process functionality and Access requirements should be shown so the user will be aware. Creating a

recursive folder to detect ransomware would be the second layer; if the ransomware passes layer one, it has complete access and can communicate with an external server. To avoid this, we need to stop the process of traversing the directories. So, we will shut it down before it gains access to other folders. In this way, preventing the encryption from happening, and it cannot encrypt without the encryption keys.

Layer 3 is about Process monitoring; this is the primary defense mechanism against ransomware. If it passes layer 2, then it has all the LocalSys Privileges. We need to monitor the behavior of the frequency of creating handlers to access directories or external communication. And the last layer will be Backup and recovery. The most important thing is the recovery of encrypted files. So, avoiding the ransom amount can be safe by having a backup of files. A proper backup in multiple locations and a quick retrieval method are significant. In their future work, they would also want to implement algorithms to detect traversing patterns followed by the processes. Concluding, Taking the required preventive actions in every layer would stop Ransomware Attacks.

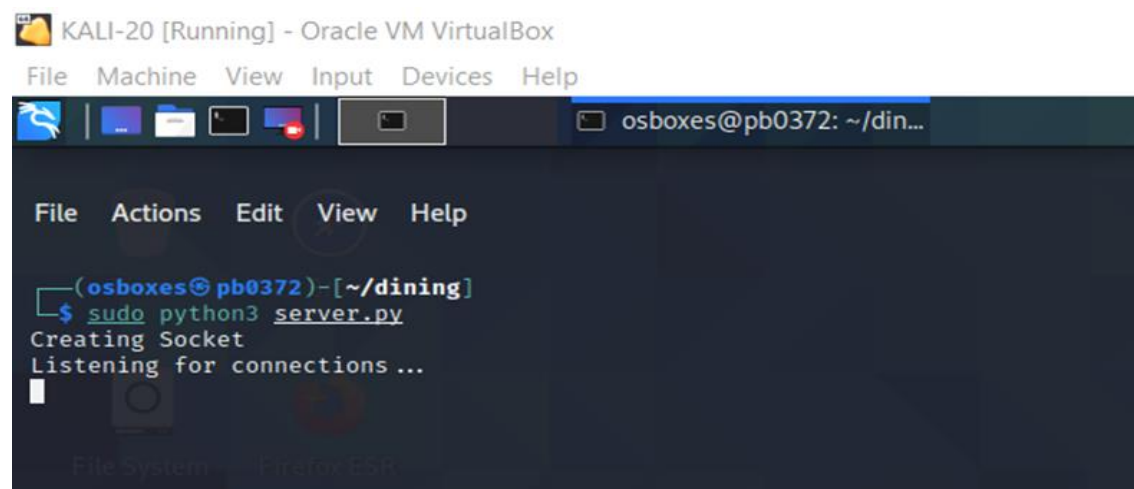
Approach

Action:

1. Python is used to implement ransomware and recursively encrypt a given directory.
2. Cryptography library- Fernet is used: It is a symmetric encryption/decryption module of the cryptography package that ensures the encrypted message cannot be manipulated/read without the key. It uses URL-safe encoding for the keys. Fernet also uses 128-bit AES.
3. Once the encryption key is generated, the concept of sockets is utilized in the encryption script to transfer the key from the victim's machine to the attacker's machine.

Attacker's machine:

1. Listening for the connection with the victim's machine using the file 'server.py'.



The screenshot shows a Kali Linux terminal window titled 'KALI-20 [Running] - Oracle VM VirtualBox'. The terminal prompt is 'osboxes@pb0372: ~/din...'. The user has entered the command 'sudo python3 server.py'. The output of the script is 'Creating Socket' and 'Listening for connections ...'. The terminal window has a menu bar with 'File', 'Actions', 'Edit', 'View', and 'Help'.

2. When the malicious executable file is run on the victim's machine and when the connection is established with the attacker's machine, the encryption key will be transferred to the attacker's machine from the victim's machine.

CSCE 5550 Fall 2022 Group 8

```
KALI-20 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

osboxes@pb0372: ~/din...

osboxes@pb0372: ~/dining
File Actions Edit View Help

(osboxes@pb0372)~[~/dining]
$ sudo python3 server.py
Creating Socket
Listening for connections...
Connection from ('10.125.248.15', 52557) established!
Connection completed and closed

(osboxes@pb0372)~[~/dining]
$ ls -lrt
total 24
-rw-r--r-- 1 root root 57 Oct 31 15:18 encrypted_host.txt
-rw-r--r-- 1 root root 291 Nov 7 18:36 decrypt_key.py_backup
-rw-r--r-- 1 root root 29 Nov 7 18:48 test.py
-rw-r--r-- 1 root root 576 Nov 7 18:58 decrypt_key.py
-rw-r--r-- 1 root root 467 Nov 30 13:52 server.py
-rw-r--r-- 1 root root 48 Nov 30 14:03 symkey.key
```

Victim's machine:

1. The directory 'Imp' on the victim's machine has 4 files and 1 subdirectory.

```
osxlab [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

Nov 30 13:01

sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp/profits
$ cd imp
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp$ ls -lrt
total 2
drwxrwx--- 1 root vboxsf 0 Nov 28 14:47 profits
-rwxrwx--- 1 root vboxsf 101 Nov 29 18:00 investment_details.txt
-rwxrwx--- 1 root vboxsf 180 Nov 30 12:47 account_details.txt
-rwxrwx--- 1 root vboxsf 143 Nov 30 12:47 billing_details.txt
-rwxrwx--- 1 root vboxsf 152 Nov 30 12:47 catalog_list.txt
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp$ head -n 3 account_details.txt
==> Account details.txt <==
Hello this file has all the information related to bank accounts involved in the business
Confidential...

==> Billing details.txt <==
This file contains everyday billing details of the dining business

Bill No      Purchaser      Seller
1            Ricky          Donald
2            Nick           Ross
3            Suraj          Shruthi

==> Catalog list.txt <==
This file contains the catalog menu details of the restaurant

Catalog ID   Name      Price
1            Salad     $9
2            Quacker   $5
3            Pasta     $8
4            Noodles   $7
5            Cookies   $3

==> Investment details.txt <==
This file has all the investment details of the business

Investor_ID   Investor_Name
1            Pra
2            Martin

==> profits <==
head: error reading 'profits': Is a directory
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp$ cd profits
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp/profits$ cat *
This file contains information about the company's profit details.
It's confidential.
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp/profits$
```

2. When the executable file 'defender' runs on the victim machine, all the files and subdirectories under the 'Imp' folder get encrypted.

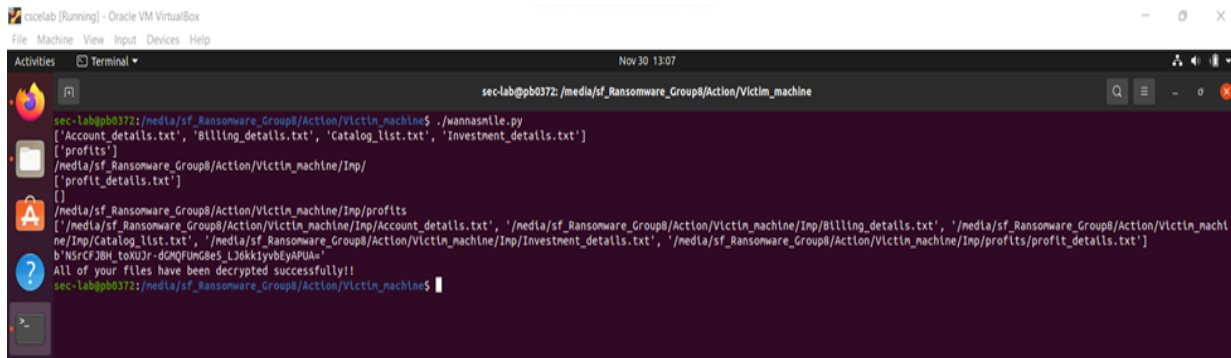
```
osxlab [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

Nov 30 13:04

sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp/profits
$ head -n 3 account_details.txt
==> Account details.txt <==
gAAAAABjh6jwCp8-Crrsck1fGz5-wkWsCs9JeroJLstIHf65-VH_9owr40y20PeasJf0j1Qz3j0vafRgI9VLEtHGDHjGdusU4vohN2g-y0Cak6qpv9G5ZKdQ2wnhp-LU3KRnchko3VwV_djYnfpvs80KxvPrngPWic2KzKzE2hgpg2eBzhtn50Hv-aHfIESBvIFkG
DvdaYp20Q2y0t2y9jKpaga==
==> Billing details.txt <==
gAAAAABjh6jwCp8-Crrsck1fGz5-wkWsCs9JeroJLstIHf65-VH_9owr40y20PeasJf0j1Qz3j0vafRgI9VLEtHGDHjGdusU4vohN2g-y0Cak6qpv9G5ZKdQ2wnhp-LU3KRnchko3VwV_djYnfpvs80KxvPrngPWic2KzKzE2hgpg2eBzhtn50Hv-aHfIESBvIFkG
DvdaYp20Q2y0t2y9jKpaga==
==> Catalog list.txt <==
gAAAAABjh6jwCp8-Crrsck1fGz5-wkWsCs9JeroJLstIHf65-VH_9owr40y20PeasJf0j1Qz3j0vafRgI9VLEtHGDHjGdusU4vohN2g-y0Cak6qpv9G5ZKdQ2wnhp-LU3KRnchko3VwV_djYnfpvs80KxvPrngPWic2KzKzE2hgpg2eBzhtn50Hv-aHfIESBvIFkG
DvdaYp20Q2y0t2y9jKpaga==
==> Investment details.txt <==
gAAAAABjh6jwCp8-Crrsck1fGz5-wkWsCs9JeroJLstIHf65-VH_9owr40y20PeasJf0j1Qz3j0vafRgI9VLEtHGDHjGdusU4vohN2g-y0Cak6qpv9G5ZKdQ2wnhp-LU3KRnchko3VwV_djYnfpvs80KxvPrngPWic2KzKzE2hgpg2eBzhtn50Hv-aHfIESBvIFkG
DvdaYp20Q2y0t2y9jKpaga==
==> profits <==
head: error reading 'profits': Is a directory
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp$ cd profits
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp/profits$ cat *
This file contains information about the company's profit details.
It's confidential.
sec-lab@pb0372: /media/xf_Ransomware_Group/Action/Victim_machine/imp/profits$
```

CSCE 5550 Fall 2022 Group 8

- Once the ransom is paid, the key is received by the victim from the attacker, and the decryption file 'wannasmile.py' along with the key executes on the victim's machine to decrypt the files and folders.



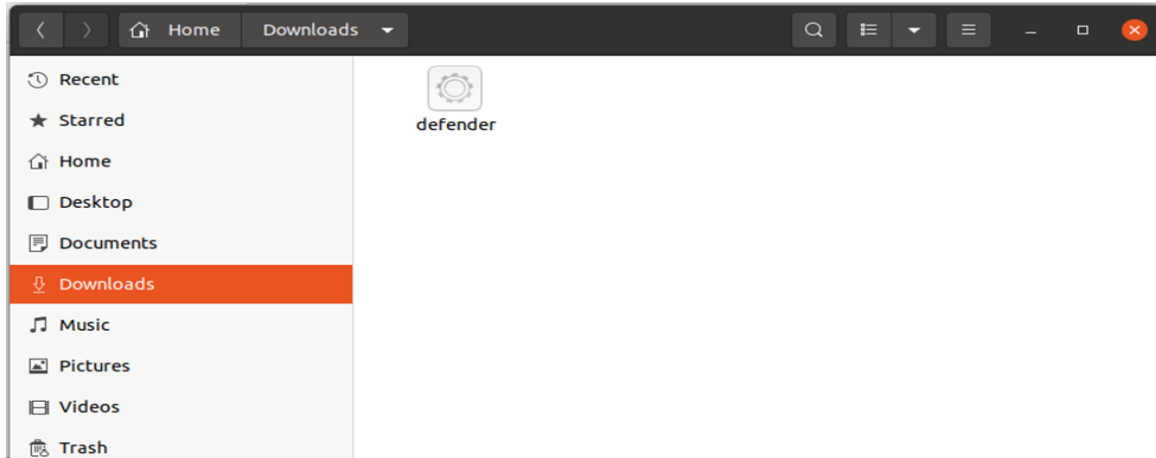
```
sec-lab@pb0372: /media/sf_Ransomware_Group8/Action/Victim_machine$ ./wannasmile.py
['Account_details.txt', 'Billing_details.txt', 'Catalog_list.txt', 'Investment_details.txt']
['profits']
/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/
['profit_details.txt']
]
/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/profits
['/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/Account_details.txt', '/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/Billing_details.txt', '/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/Catalog_list.txt', '/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/Investment_details.txt', '/media/sf_Ransomware_Group8/Action/Victim_machine/Imp/profits/profit_details.txt']
b'NSrCF38H_tokU3r-dcQPFUNG8s_L36kIyv8yAPu8'
All of your files have been decrypted successfully!!
sec-lab@pb0372: /media/sf_Ransomware_Group8/Action/Victim_machine$
```

Infection:

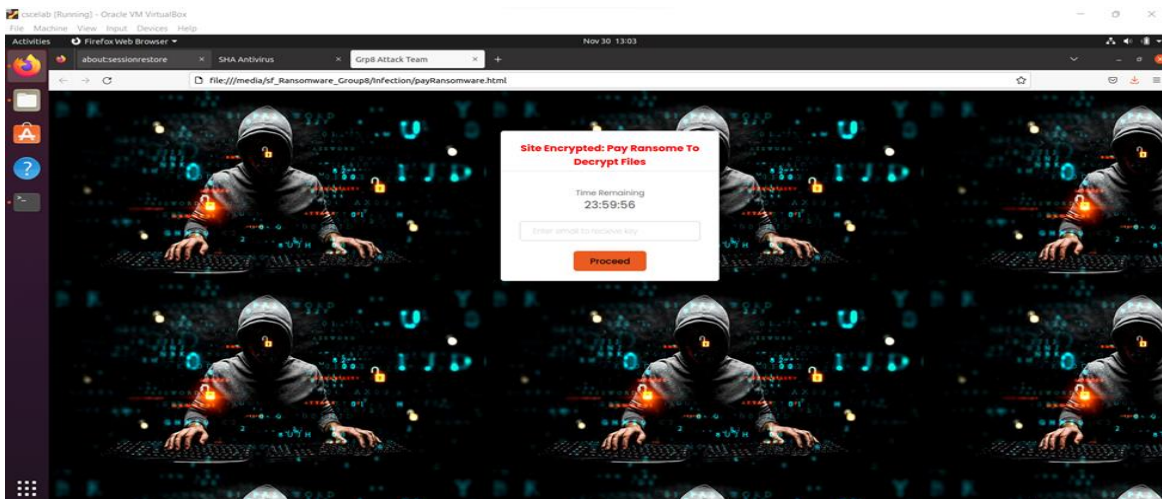
- The user will access the malicious website URL
 - An antivirus downloader website is used in this project.
 - The user is not aware that this website is malicious.
- Some actionable buttons are accessed on the webpage.
 - 'Download Now' Button is available on the malicious website to download the malicious code.



- The executable file will be downloaded.
 - Technologies such as HTML5, CSS, and JavaScript are used to create malicious websites and exploit the victim machine.



4. Malicious code gets executed via an executable File, and a new webpage will show up to the user that your system is infected post Ransomware attack.



5. The user needs to click on some action buttons to pay the amount to decrypt the infected files.
 - a. The user needs to submit an email id via form submission in the project.
 - b. The attacker shares further details about payment methods and the decryption key to decrypt the infected files using the victim's email id.

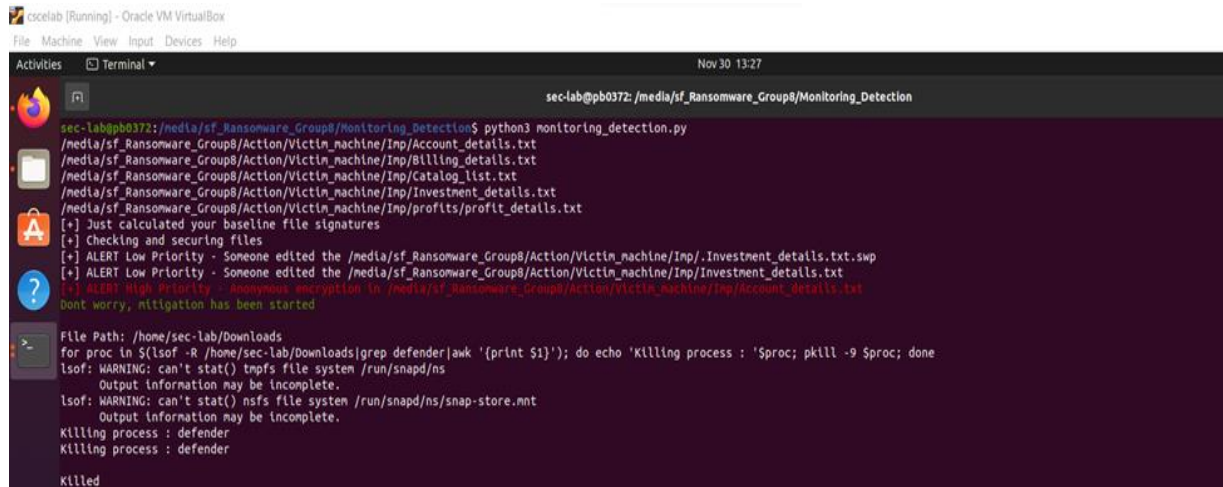
Monitoring and detection:

For monitoring, we run an infinite loop that checks all the directory files. First, we compute and compare the hash signatures of the files using hmac sha256 library functions [hmac — Keyed-Hashing for Message Authentication — Python 3.11.0 ...](#). If a file signature gets changed, the script interprets that someone edited the file. To identify if these changes are made by the legitimate user or by the encryption process, the monitoring and detection script checks the content of the file. For all the edits, alerts get triggered and displays to the user. There are two kinds of alerts:

1. Low priority alert – This is shown to the user when the legitimate users (owner) change the file by themselves. The mitigation is not currently in effect as it is a low-priority alert. In this case, the content of the file must be human-readable.

2. High Priority alert – This is shown to the user when the changes are made inside the file, and those changes are not human-readable. This raises a high-priority alert and warns the users about suspicious encryption processes. And the mitigation process starts immediately.

To identify whether the file content is human readable, we are using the "googletans" python library. [Googletans Documentation - Read the Docs](#) With the help of library functions, we can detect whether the file content is in any human-readable language. If the probability returned by the detection function is less than 50%, then the file content gets marked as encrypted (not human-readable).



```
cscelab [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Nov 30 13:27
sec-lab@pb0372: /media/sf_Ransomware_Group8/Monitoring_Detection

sec-lab@pb0372:/media/sf_Ransomware_Group8/Monitoring_Detection$ python3 monitoring_detection.py
/media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Account_details.txt
/media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Billing_details.txt
/media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Catalog_list.txt
/media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Investment_details.txt
/media/sf_Ransomware_Group8/Action/Victim_machine/Inp/profits/profit_details.txt
[+] Just calculated your baseline file signatures
[+] Checking and securing files
[+] ALERT Low Priority - Someone edited the /media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Investment_details.txt.swp
[+] ALERT Low Priority - Someone edited the /media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Investment_details.txt
[+] ALERT High Priority - Anonymous encryption in /media/sf_Ransomware_Group8/Action/Victim_machine/Inp/Account_details.txt
Dont worry, mitigation has been started

File Path: /home/sec-lab/Downloads
for proc in $(lsdf -R /home/sec-lab/Downloads|grep defender|awk '{print $1}'); do echo 'Killing process : '$proc; pkill -9 $proc; done
lsdf: WARNING: can't stat() tmpfs file system /run/snapd/ns
Output information may be incomplete.
lsdf: WARNING: can't stat() nsfs file system /run/snapd/ns/snap-store.mnt
Output information may be incomplete.
Killing process : defender
Killing process : defender

Killed
```

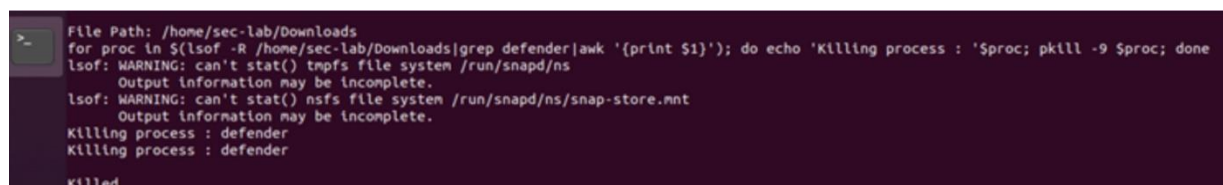
Mitigation:

Mitigation Script:

To lessen the risk of encryption of all the files, a mitigation script comes into the picture as soon as the high alert is raised.

For the mitigation script, we are giving the file path of the executable file 'defender' which is there in the download folder. Since the user has clicked on the download button on the malicious website, the latest file which got downloaded would be present in the 'Download' folder of the victim's machine.

Using the following libraries of python – [OS](#) and [System](#). We have basically written a shell script that is embedded within the python code which would identify all the processes running in the given folder, filter out just the 'defender process', and terminate it. After the mitigation script is successfully executed, it would be displayed in the command prompt as shown in the below image that the 'defender' process has been killed.



```
File Path: /home/sec-lab/Downloads
for proc in $(lsdf -R /home/sec-lab/Downloads|grep defender|awk '{print $1}'); do echo 'Killing process : '$proc; pkill -9 $proc; done
lsdf: WARNING: can't stat() tmpfs file system /run/snapd/ns
Output information may be incomplete.
lsdf: WARNING: can't stat() nsfs file system /run/snapd/ns/snap-store.mnt
Output information may be incomplete.
Killing process : defender
Killing process : defender

Killed
```

To avoid the encryption of other files, we halt the process that attempts to encrypt the files.

Backup and Recovery:

We use a shell scripting code that will copy the whole directory to the backup location at regular intervals using the Cronjob schedule.

`Cp -R * / source /destination.`

Here Cp command copies the folders, and -R is recursive, which copies the sub-directories, too, and * represents all the folders in the directory.

1. Following encryption, the user can retrieve the contents from the backup folder. If a backup can be created on different devices, it is always recommended that a backup in the cloud be taken.
2. In this case, we are using a [cronjob utility](#).
 - a. The Cron job utility is a time-based job scheduler in Unix-like operating systems. Cron allows Linux and Unix users to run commands or scripts at a given time and date. One can schedule scripts to be executed periodically. The crontab is a list of commands you want to run on a regular schedule, and the name of the command is used to manage that list.
 - b. For testing purposes, we have scheduled a python script (which is for Backup) for every 5 5 hours, and the second screenshot below is the output.

Results

According to the project findings, an unaware user visits a malicious website and downloads an executable, causing the actionable malicious code to be executed in the background and encrypt the user's data. In contrast, if a user is aware of ransomware attacks, they can use monitoring and detection techniques to prevent data breaches and effectively mitigate the attack by terminating the executable application.

To conclude, this report delves deeper into crypto-ransomware, including action, infection, monitoring, detection, and mitigation techniques for Ransome attacks.

References

[1] A NEW APPROACH TO DETECTING RANSOMWARE WITH DECEPTION YUN FENG, CHAOGE LIU, BAOXU LIU INSTITUTE OF INFORMATION ENGINEERING, CHINESE ACADEMY OF SCIENCES SCHOOL OF CYBER SECURITY, UNIVERSITY OF CHINESE ACADEMY OF SCIENCES [Poster fin \(ieee-security.org\)](#)

[2] MULTI-LAYERED DEFENSE ARCHITECTURE AGAINST RANSOMWARE MANVEER PATYAL SRINIVAS SAMPALLI DALHOUSIE UNIVERSITY, CANADA QIANG YE UNIVERSITY OF PRINCE EDWARD ISLAND, CANADA MUSFIQ RAHMAN THOMPSON RIVERS UNIVERSITY,CANADA

[3] CIMPANU, C. (2016). US SCHOOL AGREES TO PAY \$8,500 TO GET RID OF RANSOMWARE. [ONLINE] AVAILABLE AT: [HTTP://NEWS.SOFTPEDIA.COM/NEWS/US-SCHOOL-AGREES-TO-PAY-8-500-TO\[1\]GET-RID-OF-RANSOMWARE-500684.SHTML](http://news.softpedia.com/news/us-school-agrees-to-pay-8-500-to-1-get-rid-of-ransomware-500684.shtml) [ACCESSED AUGUST 2016].

[4] GAZET, A. (2010). COMPARATIVE ANⁱALYSIS OF VARIOUS RANSOMWARE VIRII. JOURNAL IN COMPUTER VIROLOGY, 6(1), PP. 77-90.

[5] GOOGLETRANS PYTHON LIBRARY DOCUMENTATION. [HTTPS://PY-GOOGLETRANS.READTHEDOCS.IO/EN/LATEST/](https://py-googletrans.readthedocs.io/en/latest/)

[6] PYTHON HMAC (KEYED HASHING AND FILE SIGNATURE) LIBRARY DOCUMENTATION

[HTTPS://DOCS.PYTHON.ORG/3/LIBRARY/HMAC.HTML](https://docs.python.org/3/library/hmac.html)

[7] RANSOMWARE DETECTION, AVOIDANCE, AND MITIGATION SCHEME: A REVIEW AND FUTURE DIRECTIONS [HTTPS://WWW.MDPI.COM/2071-1050/14/1/8/HTM](https://www.mdpi.com/2071-1050/14/1/8/HTML)

[8] Imaji, Asibi. (2019). Ransomware Attacks: Critical Analysis, Threats, and Prevention methods

[9] Maurya, A.K. & Kumar, Neeraj & Agrawal, Alka & Khan, Prof. Raees. (2018). Ransomware Evolution, Target and Safety Measures. International Journal of Computer Sciences and Engineering. 6. 80-85. 10.26438/ijcse/v6i1.8085.

[10] Hull, G., John, H. & Arief, B. Ransomware deployment methods and analysis: views from a predictive model and human responses. Crime Sci 8, 2 (2019). <https://doi.org/10.1186/s40163-019-0097-9>

[11] Connolly, Lena & Wall, David & Lang, Michael & Oddson, Bruce. (2020). An empirical study of ransomware attacks on organizations: an assessment of severity and salient factors affecting vulnerability. Journal of Cybersecurity. 6. 10.1093/cybsec/tyaa023.

[12] Cron job utility Documentation: [Linux.org](https://linux.org)

[13] Sgandurra D, Muñozgonzález L, Mohsen R, et al. Automated Dynamic Analysis of Ransomware: Benefits, Limitations and use for Detection[J]. 2016

[14] Scaife N, Carter H, Traynor P, et al. CryptoLock (and Drop It): Stopping Ransomware Attacks on User Data[C]// IEEE, International Conference on Distributed Computing Systems. IEEE, 2016:303-312.

[15] Fernet Cryptography Library: <https://cryptography.io/en/latest/fernet/>

[16] Python OS Library: <https://docs.python.org/3/library/os.html>

[17] Python System Library: <https://docs.python.org/3/library/sys.html>

[18] Finding process in a shell: <https://linuxhandbook.com/find-process-id/>

[19] Crypto-Ransomware: <https://www.techtarget.com/searchsecurity/feature/4-types-of-ransomware-and-a-timeline-of-attack-examples>

Appendix:

Link for Demo Video:

https://drive.google.com/file/d/1ZrrMS-N_PUpW9ylKvy_atpNRJOL4K_S-/view

Link for Project Code and other deliverables:

<https://github.com/abhayarora23UNT/UntComputerSecurityGroup8>
