International Journal of Engineering & Technology, 7 (2.24) (2018) 47-54

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Encryption analysis of AES-Cipher Block Chaining performance in Crypto-Wall Ransomware and SDN based mitigation

Anish Pillai[1*], M. S. Vasanthi[2], Ruturaj Kadikar[3], B.Amutha[4]

[1,2] Department of Telecommunication Engineering, SRM Institute of Science and Technology, Chennai
[3,4]Department of Computer Science Engineering, SRM Institute of Science and Technology, Chennai
[*] Corresponding author E-mail: anishpillai176@gmail.com

Abstract

The arrival of affordable high speed internet and superior computing processors has given the ability to access a novel environment of opportunities and challenges at an individuals fingertips. Sectors of education, administration, business, medical and corporate have been revolutionised by this latest onset of technology. With the arrival of IoT (Internet of Things), the number of devices that are connected to the internet will be higher than ever before. Along with this increase, the diversity of threats propagating on the internet will see a comparable increase. These threats are designed in a way to alter the integrity of the data, embed itself into other programs for further propagation and also to gain monetary compensation. In recent times, the popular malware which has made headlines worldwide is ransomware. This type of malware infection uses complex encryption of user specific files and demands a ransom for retrieval of these files. Modern ransomware behaviour indicates propagation of the malware from the host victim to other computers and shared drives within its network. This paper contains in- detailed analysis of ransomware and the currents trends of this malware. The time taken for CryptoWall ransomware encryption using AES- CBC is observed for various data sizes and a SDN approach for ransomware threat mitigation is discussed.

Keywords: *AES; Cipher Block Chaining; CryptoWall; Ransomware; Software Defined Networking.*

1. Introduction

Data storage is undergoing a drastic change in the last five years which sees local storage needs slowly migrating towards cloud based services employing technologies such as Internet of Things, vir-tualization, etc. This trend has gained momentum as it provides advantages such as rapid provisioning of resources for expansion, better redundancy, enhanced security and almost 100 percent up time. Organizations have moved their data storage into privately managed clouds for better accessibility and lower maintenance costs. The critical data files for any organization generally consist of documents such as PDF's, (Microsoft) MS Excel files, MS PowerPoint files, MS Word files, images, etc. for e.g. a hospital may create and export the medical and finance records of a patient in a PDF format for easy processing. As the movement shifts from local storage to off site storage, it faces a variety of threats which are ever evolving. The current malware trend affecting personal and commercial users is growing at a phenomenal rate and is ever changing to bypass present security measures. There is a need to have counter measures in place to mitigate and classify any threat arising for better protection and future predictability. Mitigation of any threat at network level before infection occurs can be considered as the best form of security. A interesting approach to provide network level security is exploring the security aspect of Software Defined Networking (SDN).

The year 2016 was the year in which ransomware emerged as a major security threat. Up till now, ransomware has continuously evolved itself to improve the effectiveness and propagation speed of the malware. In the last five years, individual users, data centers, companies and medical institutions are severely affected by this emerging threat. In the first quarter on 2018, a new ransomware called GandCrab known to accept ransom payment in DASH (Dash is a portmanteau of "Digital Cash") cryptocurrency affected roughly 50,000 victims. A loophole was discovered and a decryption tool was released for the first version of GandCrab. This resulted in the malware authors to release a second version in which they claimed to have secured their command and control infrastructure. An RSA algorithm is used by for encrypting victim's files, and then appends those file names by ".CRAB" and ".GDCB". The files can only be recovered by using the private key contained in the malwares command and control server. A ransom of 400 USD is asked in DASH currency which is basically moving away from standard crypto currencies such as Monero, Ethereum, Bitcoin.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

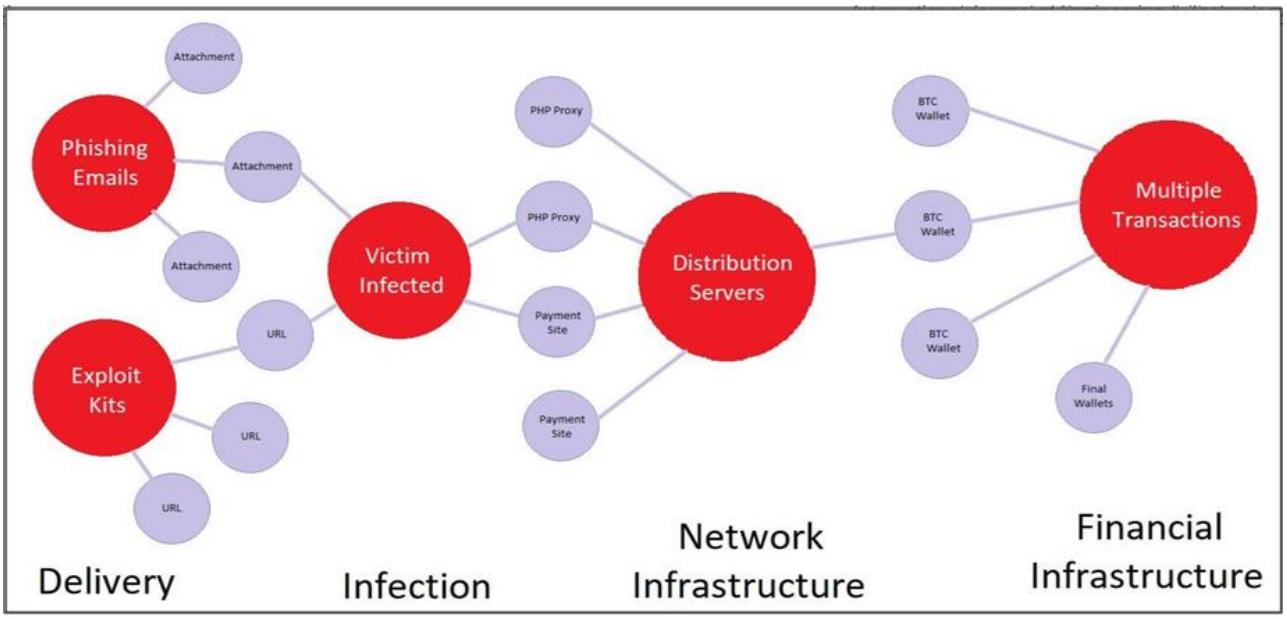


Fig. 1 : Anatomy of a CW3 attack

Internet users today are witnessing one of the most extensive ransomware campaigns known as the CryptoWall. Ransomware is a unpredictable malware that performs backend encryption on a victims files and afterwards demands a payment in exchange for the key that decrypts the said files. When a victim machine is infected, the malware targets sensitive files such as business records, databases, financial data and personal files which may hold sentimental value such as movies and photos. On identification of such files, an algo-rithm performs encryption using a key only known to the hackers. Retrieval of these files is only possible when a ransom payment is done and a decrypter key is released by them afterwards. In cases where the victim does not choose to pay the ransom, files are unlikely to be recovered if any data backup is not present. This ransomware is popular for causing irreversible damage equally to both large corporations and individual users.

Figure 1. gives the anatomy of a CryptoWall 3 attack showing the vari-ous vectors used in the ransomware mechanism. CryptoWall belongs to one of the many well known families of ransomware malware, which include TeslaCrypt, CTB-Locker, and Torrent-Locker among others. CryptoWall was first discovered by the security community in June 2014. A number of variations of CryptoWall have surfaced since then. In January 2015, the third variant (CryptoWall 3.0) was discovered infecting computers. This version of CryptoWall primar-ily distributes itself using two ways i.e. phishing emails and exploit kits. Out of the total instances of CryptoWall 3, about two-thirds of them are propagated via phishing mail. The methods used to spread CW3 is similar to those for other malware families. The filenames given to the attachments contains words such as ‘invoice’, ‘ fax’, ‘statement’, ‘bill, etc. which increase the probability of opening it. This malware not only encrypts the data on all accessible drives but can also attack personal cloud storage devices like Dropbox and shared network drives. Hence there is a possibility that a single infection can compromise multiple machines. The infected machine continues to function which allows the user to pay the ransom, as this malware does not encrypt any operating system files

Figure 2. describes the CryptoWall asymmetric Ransomware mech-anism. In Step 1, the machine is infected with ransomware using various vectors such as opening a phishing email or untrusted link. Once the machine is infected, it requests for an encryption key from the C&C (Command and Control server) server. This request is primarily a HTTP POST message which runs various data extraction scripts housed on the proxy servers belonging to the hacker

From the perspective of network traffic, CryptoWall makes use of domain names instead of IP addresses. For this purpose, a domain name service (DNS) is required. Analysis of traffic of machines infected by CryptoWall 3 indicate that it learns a victims IP address by using services which are publicly available such as curlmyip.com, myexternalip.com or ip-addr.es. The communication is primarily encrypted, and when decrypted, it is revealed that a simple text protocol is used to reports information of the infected machine like IP address, unique identifier, number of files present, MAC address, etc. to the malware owned C&C server. In step 2 once the C&C server acknowledges this request, it generates two keys to complete the process of infection. Step 3 is where the public key is sent back to the infected machine for encryption. The other is the private key which is never shared by the attacker and is used for decryption of the data after the ransom is paid. The infected machine is then displayed with an image showing instructions on how the ransom is to be paid. This shows that if the above scenario is correctly implemented, it is practically impossible to break crypto ransomware [5].

Software defined networking (SDN) is a networking model that has gained popularity which can overcome the drawbacks of the current network frameworks. SDN can be defined as a network architecture in which the control and the data planes are decoupled thereby making network devices simple forwarding elements. Figure 3 gives the difference between traditional networking and Software Defined Networking

. The forwarding decisions are based on flow instead of destination which is basically a group of packet field values acting like a filter criteria for a set of corrsponding actions.

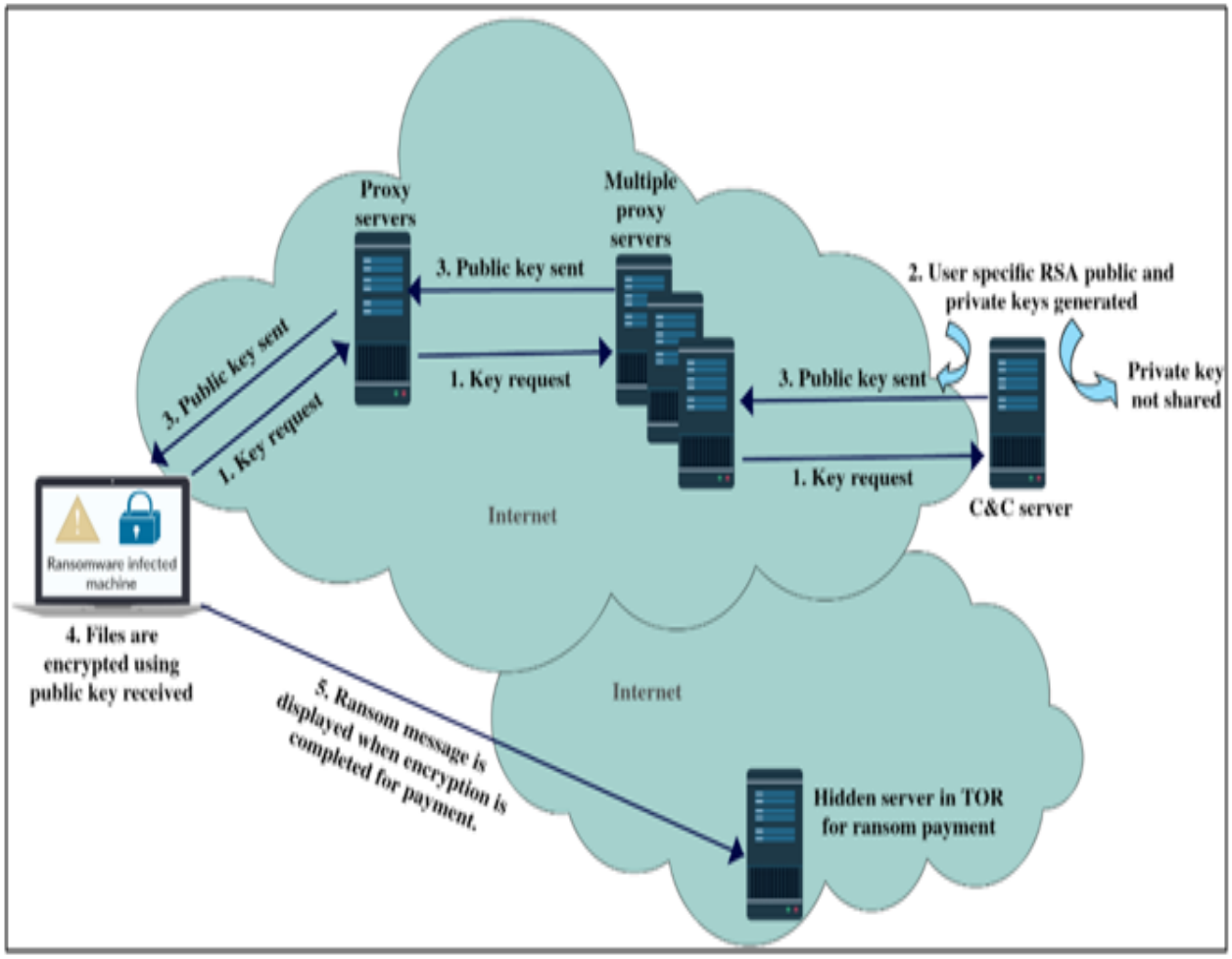


Fig. 2: Crypto locker asymmetric ransomware

SDN can be simply visualized as a software application running on propriety hardware on a logically centralized view that programs network functions by interacting with the underlying data plane. As the control and data plane is decoupled in SDN, the underlying network infrastructure is separated from the applications, and hence the network can be managed logically in a centralized way which simplifies policy enforcement and network reconfiguration.

OpenFlow has literally become the standard that is used to implement the SDN model. This protocol enables networking devices like routers and switches to be handled by an external controller, which have dependency on the internal flow tables. The networking devices process all the packets by comparing it to its flow table (and all consisting flow entries). Based on the outcome, action is taken for every matched flow entry or are forwarded to the controller in case no match is found. Hence, real time application of the traffic control rules is possible. Network security can be provided more efficiently and flexibly using SDN by considering the following example: when a host is detected to be performing malicious behavior, the SDN controller can update the control policies and rules immediately of its networking devices. Current research in the field of SDN security to mitigate ransomware threats is limited. Present counter measures are deployed only after companies analyze and release software updates for enhanced protection which may take a long cycle of time. In this paper, we analyze the encryption used for Crypto wall ransomware and discuss about using SDN as a successful method for mitigating it.

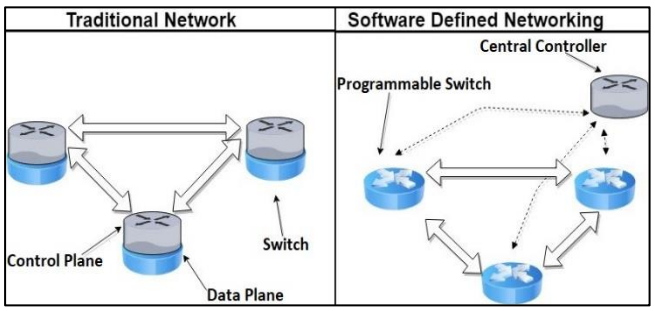


Fig. 3: Traditional Networks vs SDN

CryptoWall employs AES-CBC for encryption of data in a system. Among the several techniques available, Advanced Encryption Standard (AES), the Data Encryption Standard (DES) and Triple DES (3DES) are the most popular used. Out the three techniques motioned above, AES is preferred due to the advantages it provides. AES is made of three types namely AES-128, AES-192 and AES- 256. 128 bits of data is encrypted and decrypted using key of size 128, 192 and 256 bits. 128 bit keys undergo 10 rounds, 192 bit key undergoes 12 rounds and 256 bit keys undergo 14 rounds

of processing steps which basically include substitution, transposition and jumbling of the plaintext to convert it into cipher text. AESCipher Block Chaining (CBC) is chosen over AES- Electronic Code Book (ECB) [27] as it does not contain the weakness of near obvious data pattern found in ECB. In AES-CBC encryption according to Figure 4 (a), each plaintext block is XORed with the previous cipher text block before it is encrypted. Hence the name 'chaining' is used. An Initialization Vector (I.V) which is securely generated and random in nature is used to XOR with the first plain text block. Figure 4 (b) shows the decryption process. The cipher text of the previous block is XORed with the decrypted text to give the plain text. The I.V used for decrypting the first block is same as one used for encryption.

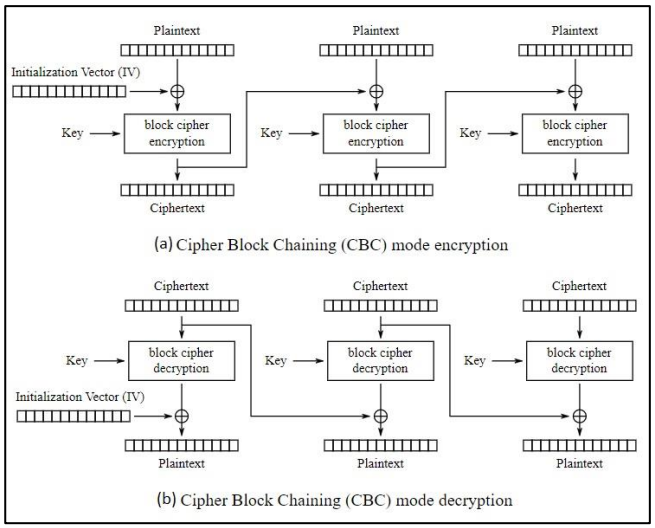


Fig. 4 :AES- Cipher Block Chaining

The rest of the paper is organized as follows. Section II provides Motivation for choosing ransomware as a topic of research. Section III presents the Literature Survey, followed by Implementation done to observe the results in Section IV. Section V consists of Conclusion followed by Acknowledgement.

2. Motivation

In the last five years, ransomware attacks have increased exponentially. The mechanism of ransomware being simple yet so effective makes it a threat that is difficult to mitigate. With the availability of high speed internet and limited understanding of ransomware, this malware has affected computing users across the globe. Although security patches and updates have been released by proprietary companies for various operating systems for protection against ransomware, they all deal with the problem once it has successfully infected enough machines to gain profits. In case of the modern ransomware, asymmetric encryption is used where two separate keys are used for encryption and decryption. The public key is used to encrypt the data and the private key is stored within the C&C server for decryption of data once the payment is made. The aim to develop a method to mitigate the threat before the encryption process completes gives a fighting chance to mitigate ransomware. Ransomware in the first quarter of 2016 extorted $6 million from various businesses according to the FBI. If we consider the operational downtime, replacement of computer systems and increases resources used to resolve the problem, ransomware has actually cost businesses more than $75 billion dollars in damages. A study carried out by IBM security states that ransomware infected mails increased by 6,000 percent as compared to the previous year. It also states that almost 40 percent of all spam messages, i.e. one of every two spam emails is infected with a ransomware. A survey by Datto, a data protection company in 2016 revealed that 92 percent of the total 1,100 IT firms surveyed had clients affected by ransomware and almost 40 percent of the attacks occurred in the latter half of 2016. According to statistics released by Symantec, 100 more malware families have been identified which is more than triple the amount seen previously which is an alarming 36 percent increase. As compared to the world, USA was the biggest target as 64 percent Americans are prepared to pay the ransom compared to 34 percent global affected users. The average ransom demanded is approx. $544 which indicates that attackers have finally found out the ideal amount which ensures highest ransom payments. In 2017, USA remains the country with the highest number of attacks (29%), followed by Japan (9%), Italy (8 %), India and Germany (4 %) and Netherlands, United Kingdom, Australia, Russia and Canada each at 3%. A SNS Research report estimates that between 2016 and 2020 investments by service providers in SDN/NFV will grow at a Compound Annual Growth Rate (CAGR) of 46%, which will account for over USD 18 billion in revenue.

A method is proposed to mitigate ransomware using SDN by detecting malicious packet flows between infected machine and attacker server. The time required for encrypting an infected system provides higher detection rate of malicious packet flows for a SDN controller. Although the system might be fully compromised, a sacrifice of one system to protect the entire network is a good bargain in mitigating the threat. In a data center, any particular server infected with ransomware can bring down network. Implementing strict rules and policies for malicious traffic detection provides better security.

3. Literature survey

Within the first quarter of 2018, the internet has crossed over 3.8 billion users. Segovia et al. [1] states that by using email as vector, malicious infections are popularly as they have a higher probability of being opened upon arrival. The papers states that sympathy attacks have greater effectiveness than intimidation attacks using email as victims choose to download the software because they felt rewarded for their work.

L. Munoz-gonz alez et.al gives the chronological order of ransomware which have been discovered in the last 3 decades along with its features in Figure 5 [2]. Adamov et al. [3] focuses on the characteristics of ransomware such as delivery method, file type, platform, files encryption method, encryption locations, communication with C&C server, decryption service location, payment information, target audience, etc. Cabaj et al. [4] has concluded that ransomware using asymmetric keys for encryption typically take 4 to 30 seconds to download the public encryption key from the C&C server. They use an SDN approach for blocking ransomware communication between the victim system and C&C server before the public key is downloaded thereby stopping ransomware from encrypting the victims system.

Caivano et al. [5] analyses a dataset consists of 76 samples belonging to twelve families: TeslaCrypt, Cryptolocker, CryptoWall, CTB Locker, 'los pollos hermanos', CryptoFortress, DirCrypt,

EncryptorRaas, Toxic, ZeroLocker, Winlock, ACCDFISA and comes to the conclusion that four families out the twelve encrypt the local drives only but the rest of the families repeat the encryption process on all reachable drives connected to the infected machine. This is the methods which will be used by future versions of ransomware. Only four families i.e. Cryptowall, TeslaCrypt, CTB-locker, and CryptoFortress delete the original copies which restricts the victim to rebuild previous versions of the files and to disable recovery in order to refrain the user from recovering a part of the infected files, Cryptowall disables the system restore, for the same reason. Chen et al. [6] develops a method for characterization of Android ransomware by collecting and analyzing 2,721 samples. A innovative system called RansomProber was designed for abnormal encryption detection which helps to detect and stop the encryption process before the files are lost forever. Kamdar et al. [7] focuses on the intricate working of crypto ransomware by analyzing ransomware work flow, behavior and mitigation strategies in order to better understand it. Chen et al. [8] puts forward a method in which the features of various malware can be automatically extracted using host logs. This method uses a Term Frequency-Inverse Document Frequency (TF- IDF) metric for analyzing malware based on Fisher's Linear Discriminant Analysis.

Name	Year	Notable Features
PC Cyborg	1989	Spreads using floppy disks
GPCoder	2005-2008	Spreads via emails; encrypts a large set of files
Archiveus	2006	First Ransomware to use RSA encryption
WinLock	2010	Blocks PCs by displaying a ransom message
Reveton	2012	Warning purportedly from a law enforcement agency
DirtyDecrypt	Summ. 2013	Encrypts eight different file formats
CryptLocker	Sept. 2013	Fetches a public key from the C&C
CryptoWall	Nov. 2013	Requires TOR browser to make payments
Android Defender	2013	First Android locker-ransomware
TorDroid	2014	First Android crypto-ransomware
Critroni	July 2014	Similar to CryptoWall
TorrentLocker	Aug. 2014	Stealthiness: indistinguishable from SSH connections
CTB-Locker	Dec. 2014	Uses Elliptic Curve Cryptography, TOR and Bitcoins
CryptoWall 3.0	Jan. 2015	Uses exclusively TOR for payment
TeslaCrypt	Feb. 2015	Adds the option to pay with PayPal My Cash Cards
Hidden Tear	Aug. 2015	Open source ransomware released for educational purposes
Chimera	Nov. 2015	Threatens to publish users' personal files
CryptoWall 4.0	Nov. 2015	Encrypts also filenames
Linux.Encoder.1	Nov. 2015	Encrypts Linux's home and website directories
DMA-Locker	Jan. 2016	Comes with a decrypting feature built-in
PadCrypt	Febr. 2016	Live Chat Support
Locky Ransomware	Febr. 2016	Installed using malicious macro in a Word document
CTB-Locker for WebSites	Febr. 2016	Targets Wordpress
KeRanger	Mar. 2016	First ransomware for Apple's Mac computers
Cerber	Mar. 2016	Offered as RaaS (& quote in Latin)
Samas	Mar. 2016	Pentesting on JBOSS servers
Petya	Apr. 2016	Overwrites MBT with its own loader and encrypts MFT
Rokku	Apr. 2016	Use of QR code to facilitate payment
Jigsaw	Apr. 2016	Press victims into paying ransom
CryptXXX	May 2016	Monitors mouse activities and evades sandboxed environment
Mischa	May 2016	Installed when PETYA fails to gain administrative privileges
RAA	June 2016	Entirely written in Javascript
Satana	June 2016	Combines the features of PETYA and MISCHA
Stampado	July 2016	Promoted through aggressive advertising campaigns on the Dark web
Fantom	Aug. 2016	Uses a rogue Windows update screen
Cerber3	Aug. 2016	Third iteration of the Cerber ransomware

Fig. 5: Chronological order of ransomware types

Silva et al. [9] has proposed a model for detection and prevention of ransomware such as Petya and WannaCry by detecting the main vulnerabilities in their behavioral pattern. These vulnerabilities are then used in data analysis which are extracted using automatic learning algorithms. Paper by Chadha et al. [10] states that several machine learning algorithms have been assessed for discovering domain generating algorithms which play a vital role in the spread of ransomware such as CryptoWall. Sanatinia et al. [11] focuses on the working of OnionBots using infrastructures like TOR (The Onion Router) which play a vital role in providing anonymity to hackers and illegal activities like ransomware. Mitigation methods are proposed at Tor level for this OnionBots which are able to self heal even if 90% of the nodes are deleted. Bhattacharya et al. [12] proposes cloud architecture based on practical implementation and methods which provide a 90% robustness in the face of a ransomware attack. Cabaj et al. [13] proposes a innovative system for ransomware detection that analyses the malware communication characteristics of two ransomware families namely, Locky and CryptoWall. By analyzing the HTTP messages between the victim system and the C&C server, the nature of the communication can be classified as an malicious flow. The experimental results were able to achieve detection rates of 97-98%. Thus detection of such flows can result in blocking malicious packets thereby stopping ransomware to fully deploy in an victims machine. Shinde et al. [14] states since ransomware is a new topic of threat which was previously unheard of, elder people have less awareness regarding it. The papers showcases that strategies are already employed in place to mitigate threats such as ransomware, but the awareness on the usage of such strategies is not known to all. The paper also confirms that the mindset about ransomware is such that, all victims are unwilling or unable to pay the ransom amount due to limited knowledge on nature of the payment method i.e. cryptocurrencies.

Kreutz et al. [15] concludes that traditional networks are tedious to manage as the control and data planes are integrated vertically and are presently vendor specific. Also, these traditional devices have their own line of products which may haves its own management interfaces and configurations meaning product updates and upgrades take a longer period of time. Change and innovation is severely restricted as the devices are vendor specific. Gopi et al. [16] publishes that Software-Defined Networking (SDN) in recent years has been a main focus of security research due to the various advantages it offers over traditional networking. SDN will replace traditional networking as a promising and robust network architecture, as it brings ease of network management concerning programmability, simplicity, and elasticity. The author I. Ahmad et.al [17] highlights the vulnerabilities and security threats to the control, data and application planes of SDN infrastructure. Security solutions to strengthen the network security according to ITU-T in SDN is summarized in this paper. It states that since the central controller is the most vulnerable component in a SDN architecture, its vulnerability has been tested for various scenarios. The author also highlights the issue that although development and implementation of the security applications is done via the application plane, the security of the application plane itself is a major security challenge. Z. Shu et.al [18] focuses on the various security aspects provided by SDN by analyzing the threats and countermeasures in great detail from three perspectives, i.e., the control layer, the data forwarding layer, and the working application. They have analyzed four threats which compromises the security of an SDN infrastructure. The counter measures proposed are for the following: (1) Man-in-middle attack between switch and controller. (2) Denial of Service (DoS) attack to saturate the flow table and flow buffer. (3) DoS/DdoS attack on controller and (4) Threats on distributed multi controllers. A. Feghali et.al [19] states that while designing the architecture and protocols, the security solutions must be embedded within it. This will result in efficient and smooth migration to SDN as a complete secure solution is already defined. Dabbagh et al.[20] mentions the advantages which makes it a suitable technology for controlling networks by providing a logically centralized architecture with improved security aspects. A anomaly detection system based on SDN was put forward by Mehdi et al. in 2011 [21]. Zaalouk et al. [22] utilized SDN to de-

tect the various attack on a network and the malicious behavior of software on mobile devices was put forward by R. Jin et.al [23]. S.Shin and G.Gu [24] propose a framework called CLOUD-WATCHER to provide monitoring services for a cloud network which are dynamic and large in nature. This framework functions by automatically detouring network packets for inspection using network security devices which are pre installed. In addition to this, the operations can also be implemented by writing a simple policy script which allows the cloud administrator to enhance the protection provided. Yost et al. [25] proposes a new method called as malFire which is basically a high level firewall designed for purpose of advertisement blocking and malware detection. A. McNeil states WannaCry ransomware spreads using the alleged NSA-leaked EternalBlue exploit to gain access into the network by tracking down vulnerable public facing SMB ports. This ransomware then installs by using the DoublePulsar exploit which is also a alleged NSA leak. The Advanced Encryption Standard (AES), is a block cipher used as a standard by the U.S. government for government and military use. AES- ECB (Electronic Codebook) is the most basic form of block cipher encryption. CBC (Cipher Blocker Chaining) is an advanced form of block cipher encryption. With CBC encryption, each cipher text block depends on all plain text blocks which are processed up to that point. AES [26] is said to be a symmetric key algorithm as the same key is used for encryption as well as decryption.

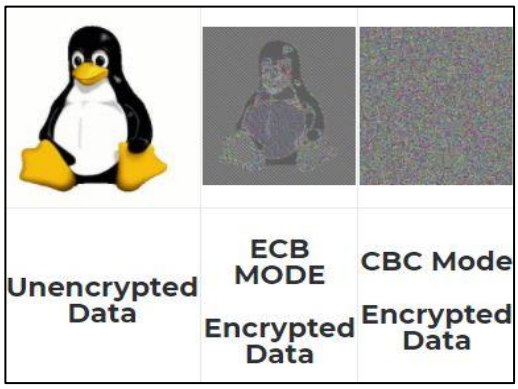


Fig. 6: Difference between AES-EBC and AES-CBC 9

The data block length in AES [27] is defined to 128 bits and the key sizes are of 128, 192 and 256 bits. AES is an iterative algorithm and each iteration performed is called a round. There can be 10, 12 or 14 rounds which depends on the key sizes chosen i.e. 1128, 192 or 258 bits. Every round except the final round in AES is made up of four transformations namely Sub Bytes, ShiftRows, MixColumns, and AddRoundKey. MixColumns transformation is not included in the final round. Decryption process is the reverse of the encryption process and each operation is the inverse of the equivalent one in the encryption process.

4. Implementation and observation

Table 1: System Specifications of Machines Performing Encryption

Processor	Intel Core i7-6700 CPU @ 3.40 GHz
Cores	8
Storage	500 GB
O.S	Ubuntu 14.04 LTS

The code is written in Python as an script to encrypt drives on a computer. The same script can be run on Windows based operating systems to achieve similar results. In order to observe and study the encryption process, the mechanism used is Advanced Encryption Standard- Cipher Block Chaining (AES-CBC) using symmetric encryption as the data needs to be decrypted as well. The key size is varied as a 16 byte, 24 byte and a 32 byte key which corresponds to 128, 192 and 256 bits of key respectively. The different data sizes taken for encryption is 30 GB, 100 GB, 200 GB and 300 GB. The file type encrypted is documents, photos, executable files and all non OS extension files. Large file extensions such videos and movies of sizes greater than 1 GB are skipped as encrypting such files will taker longer than usual. The data file size which has been considered is the standard size of user files in a normal system of any individual. The encryption is run on a block of different data sizes accordingly and the time corresponding to each encryption cycle is noted. After each encryption cycle, the system is rebooted to refresh all settings for obtaining accurate results. The files are then decrypted by using a script which works in an inverse way to decrypt the data.

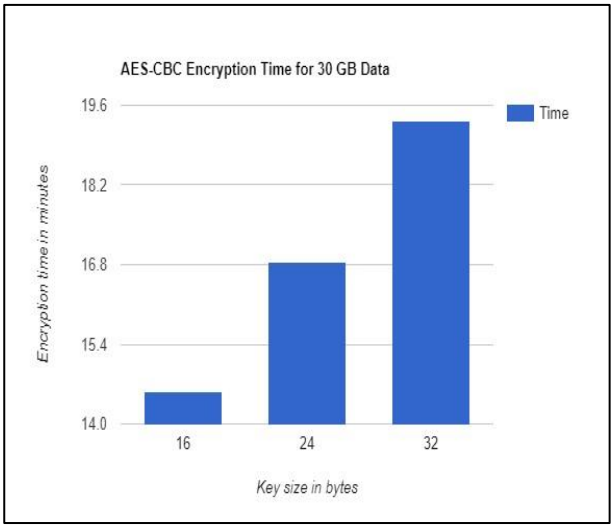


Fig. 7: AES-CBC encryption time for 30 GB data

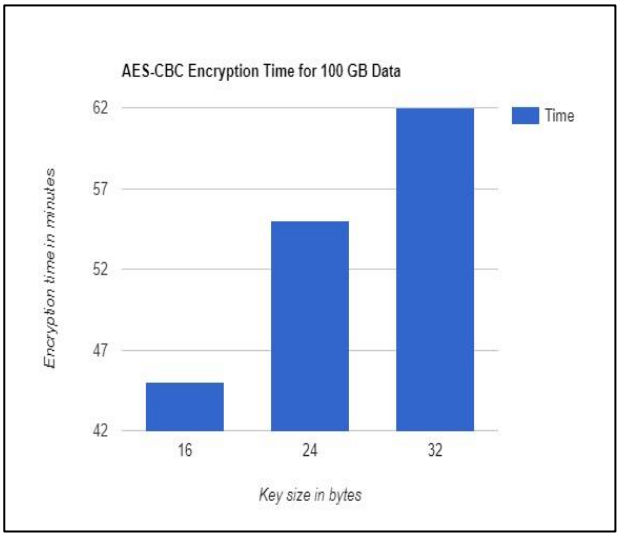


Fig. 8 : AES-CBC encryption time for 100 GB data

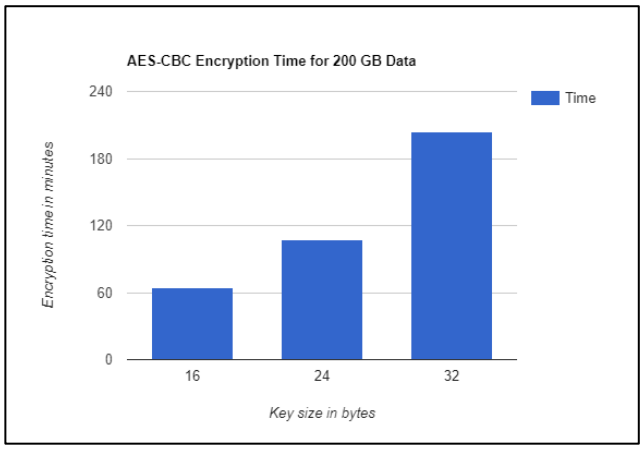


Fig. 9: AES-CBC encryption time for 200 GB data

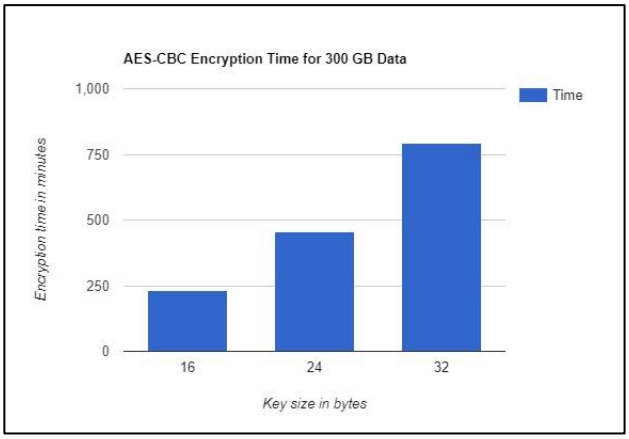


Fig. 10: AES-CBC encryption time for 300 GB data

The graphs show the time taken for encrypting various data sizes. Figures 7 to 10 gives the time analysis for data sizes from 30 GB to 300 GB. It is observed from the graphs that the time taken for encryption increases as the key size increases. A greater key size chosen gives higher encryption quality. Also, we can clearly come to an conclusion that the encryption time is directly proportional to the data size. Since CryptoWall 3 makes use of AES-CBC for encryption, the time taken is relatively higher as compared to other ransomwares.

5. Conclusion

Ransomware is a security threat that is thriving over time and solutions to thoroughly mitigate it are still under research. Attackers are persistently devising new techniques to propagate this malware for catastrophic situations. In this work, encryption time is observed for different key and data sizes to analyze the processes running in background after initialization of ransomware infection. Infection time is dependent on system configuration, key size and data size. With the increase in the number of devices connected to the internet, ransomware is also observed to affect mobile devices apart from servers and desktops. Notably, the deciding feature in employing counter measures for ransomware is the time taken to detect and react to the malicious packet flows. Hence in future, this work will be extended to include SDN paradigm to analyze malicious packet flows for restricting communication between ransomware script and attacker servers. Moreover, a coordinated database of malicious packet flows will be maintained between multiple SDN controllers distributed over a geographic region to strengthen ransomware defense. More importantly, to efficiently fight ransomware, it is very important to break the business model of malware developers. If the number of infections decrease, the revenue generated for the malware developers decrease which in turn increases the operating cost for the upkeep of infrastructure. Presently, the best method to recover from a ransomware attack is to restore data from a weekly or monthly backup taken in an off line storage medium.

6. Acknowledgement

We would like to express our gratitude to the research group, Software Defined Research Lab, Department of Computer Science Engineering, SRM Institute of Science and Technology, Kattankulathur, Chennai, for their kind assistance and providing us with the required resources.

References

[1] P. L. Gallegos-Segovia, J. F. Bravo-Torres, V. M. Larios-Rosillo, P. E. Vintimilla-Tapia, I. F. Yuquilima-Albarado and J. D. Jara-Saltos, “Social engineering as an attack vector for ransomware,”, 2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Pucon, 2017, pp. 1-6.

[2] Sgandurra, Daniele, Luis Mu˜noz-Gonz´alez, Rabih Mohsen, and Emil C. Lupu., “Automated dynamic analysis of ransomware: Benefits, limitations and use for detection.”, Proceedings of the conference name, arXiv preprint arXiv:1609.03020 (2016).

[3] A. Adamov and A. Carlsson, “The state of ransomware. Trends and mitigation techniques,”, 2017 IEEE East-West Design & Test Symposium (EWDTS), Novi Sad, 2017, pp. 1-8.

[4] K. Cabaj and W. Mazurczyk, “Using Software-Defined Networking for Ransomware Mitigation: The Case of CryptoWall,”, IEEE Network, vol. 30, no. 6, pp. 14-20, November-December 2016.

[5] D. Caivano, G. Canfora, A. Cocomazzi, A. Pirozzi and C. A. Visaggio, “Ransomware at X-Rays,”, 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Exeter, 2017, pp. 348-353.

[6] J. Chen, C.Wang, Z. Zhao, K. Chen, R. Du and G. J. Ahn, “Uncovering the Face of Android Ransomware: Characterization and Real-Time Detection,”, IEEE Transactions on Information Forensics and Security,, vol. 13, no. 5, pp. 1286-1300, May 2018.

[7] Neha Kamdar, Vinita Sharma, Abhishek Sengar and Rahul Tiwari, “Detection and prevention of Rreq flooding attack through filtering scheme”, International Journal of Research In Technology and Management (IJRTM), Volume 3 Issue 3, June 2017.

[8] Qian Chen, Robert A. Bridges, “Automated Behavioral Analysis of Malware A Case Study of WannaCry Ransomware ”, 2017 16th IEEE International Conference on Machine Learning and Applications, 2017, pp. 454-460.

[9] J. A. H. Silva and M. Hern´andez-Alvarez, “Large scale ransomware detection by cognitive security,”, 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM), Salinas, 2017, pp. 1-4.

[10] S. Chadha and U. Kumar, “Ransomware: Let’s fight back!,”, 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida,, 2017, pp. 925-930.

[11] Amirali Sanatinia, Guevara Noubir, “OnionBots: Subverting Privacy Infrastructure for Cyber Attacks”, 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 2015, pp.69-80.

[12] S. Bhattacharya and C. R. S. Kumar, “Ransomware: The CryptoVirus subverting cloud security,”, 2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), Chennai, 2017, pp. 1-6.

[13] Cabaj Krzysztof, Gregorczyk Marcin, MazurczykWojciech, “Softwaredefined networking-based crypto ransomware detec-

tion using HTTP traffic characteristics", Computers and Electrical Engineering, 2017, pp. 1-16.
[14] R. Shinde, P. Van der Veeken, S. Van Schooten and J. van den Berg, "Ransomware: Studying transfer and mitigation,", 2016 International Conference on Computing, Analytics and Security Trends (CAST), Pune, 2016, pp. 90-95.
[15] D. Kreutz, F. M. V. Ramos, P. E. Ver´ıssimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-Defined Networking: A Comprehensive Survey,", Proceedings of the IEEE, ol. 103, no. 1, pp. 14-76, Jan. 2015.
[16] D. Gopi, S. Cheng and R. Huck, "Comparative analysis of SDN and conventional networks using routing protocols,", 2017 International Conference on Computer, Information and Telecommunication Systems (CITS), Dalian, 2017, pp. 108-112.
[17] I. Ahmad, S. Namal, M. Ylianttila and A. Gurtov, "Security in Software Defined Networks: A Survey,", IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2317-2346, Fourthquarter 2015.
[18] Zhaogang Shu, Jiafu Wan, Di Li, Jiaxiang Lin, Athanasios V Vasilakos, Muhammad Imran, "Security in Software-Defined Networking: Threats and Countermeasures", Mobile Networks and Applications, Vol.21, No.5, (2016), pp.764-776.
[19] A. Feghali, R. Kilany and M. Chamoun, "SDN security problems and solutions analysis,", 2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS), Paris, 2015, pp. 1-5.
[20] M. Dabbagh, B. Hamdaoui, M. Guizani and A. Rayes, "Softwaredefined networking security: pros and cons,", IEEE Communications Magazine, vol. 53, no. 6, pp. 73-79, June 2015.
[21] Mehdi, Akbar Syed, Junaid Khalid, Syed Ali Khayam, "Revisiting Traffic Anomaly Detection Using Software Defined Networking", Proceedings of the conference name, Springer Berlin Heidelberg (2011), pp.161-180.
[22] Mehdi, Akbar Syed, Junaid Khalid, Syed Ali Khayam, "Revisiting Traffic Anomaly Detection Using Software Defined Networking", Proceedings of the conference name, Springer Berlin Heidelberg (2011), pp.161-180.
[23] R. Jin and B. Wang, "Malware Detection for Mobile Devices Using Software-Defined Networking,", 2013 Second GENI Research and Educational Experiment Workshop, Salt Lake City, UT, 2013, pp. 81- 88.
[24] Seungwon Shin and Guofei Gu, "CloudWatcher: Network security monitoring using OpenFlow in dynamic cloud networks (or: How to provide security monitoring as a service in clouds?),", 012 20th IEEE International Conference on Network Protocols (ICNP), Austin, TX, 2012, pp. 1-6.
[25] W. Yost and C. Jaiswal, "MalFire: Malware firewall for malicious content detection and protection,", 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), New York, NY, 2017, pp. 428-433.
[26] M. Vaidehi and B. J. Rabi, "Design and analysis of AES-CBC mode for high security applications,", Second International Conference on Current Trends In Engineering and Technology - ICCTET 2014, Coimbatore, 2014, pp. 499-502.
[27] G. C. Prasetyadi, A. Benny Mutiara and R. Refianti, "File encryption and hiding application based on advanced encryption standard (AES) and append insertion steganography method,", 2017 Second International Conference on Informatics and Computing (ICIC), Jayapura, 2017, pp. 1-5.
[28] S.V. Manikanthan , T. Padmapriya "An enhanced distributed evolved node-b architecture in 5G tele-communications network" International Journal of Engineering & Technology (UAE), Vol 7 Issues No (2.8) (2018) 248-254.March2018
[29] S.V.Manikanthan and T.Padmapriya "Recent Trends In M2m Communications In 4g Networks And Evolution Towards 5g", International Journal of Pure and Applied Mathematics, ISSN NO:1314-3395, Vol-115, Issue -8, Sep 2017.
[30] S.V. Manikanthan, T. Padmapriya, Relay Based Architecture For Energy Perceptive For Mobile Adhoc Networks, Advances and Applications in Mathematical Sciences, Volume 17, Issue 1, November 2017, Pages 165-179