Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 94 (2016) 465 – 472

Procedia Computer Science

The 2nd International Workshop on Future Information Security, Privacy & Forensics for Complex Systems

(FISP 2016)

Experimental Analysis of Ransomware on Windows and Android Platforms: Evolution and Characterization

Monika[a]*, Pavol Zavarsky[a], Dale Lindskog[a]

[a]*Information System Security Management, Concordia University of Edmonton, Edmonton, T5B 4E4, Canada*

Abstract

The focus of the paper is on providing insights on how ransomware have evolved from its starting till March 2016 by analyzing samples of selected ransomware variants from existing ransomware families in Windows and Android environments. Seventeen Windows and eight Android ransomware families were analyzed. For each ransomware family, at least, three variants belonging to the same family were compared. The analysis revealed that ransomware variants behave in a very similar manner, but use different payloads. Our analysis shows that there has been a significant improvement in encryption techniques used by ransomware. The experimental results in Windows environment demonstrate that detection of ransomware is possible by monitoring abnormal filesystem and registry activities. In Android environment, our analysis reveals that likelihood of ransomware attacks can be reduced by paying a closer attention to permissions requested by the Android applications.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Ransomware; Evolution; Characterization; Payload; Encryption; Permissions; Windows; Android;

* Corresponding author. Tel.:+ 17802327187 .
E-mail address: monikachoudhary24@gmail.com

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.08.072

1. Introduction

Ransomware is a form of malicious code or malware that infects a computer and spreads rapidly to encrypt the data or to lock the machine. This malware makes the data inaccessible to the users and the attackers demand payment from the user in order to have their files unencrypted and accessible. The payment is often requested in Bitcoin[1] or other untraceable currency.[2] Businesses and individuals worldwide are currently under attack by ransomware. The main purpose of ransomware is to maximize the monetization using malware. Looking from its first infection till now, ransomware has shown its disruptive and destructive side. It has started doing a lot more than just displaying advertisements, blocking services, disabling keyboard or spying on user activities. It locks the system or encrypts the data leaving victims helpless to make a payment and sometimes it also threatens the user to expose sensitive information to the public if payment is not done. All families of ransomware behave in an almost similar manner but uses different kinds of payload. This paper focuses on providing the insight view on how ransomware works and the way it has evolved during this time period by observing some samples related to these selected ransomware families. As earlier ransomware used any software's expired license as a trick to ask for payment in the form of license renewal. Now, attackers try to lock the system access, block the use of mouse and keyboard, encrypt files and now these attackers try to act as law enforcement agencies. This paper explains how the ransomware samples are functioning and how our findings can be used to identify ransomware.

Section II of this paper discusses the Ransomware sample set, its collection and naming scheme. Section III presents the analysis results from Anubis, Andrubis[3], and Cuckoo Sandbox[4]. This section also expresses the life cycle of ransomware on both Windows and Android environments. Section IV elaborates the evolution of ransomware throughout these years. Section V presents few detection techniques which can be used to verify the presence of ransomware. Section VI concludes by highlighting results and findings from this work and the future work that can be done in regards with ransomware.

2. Ransomware Dataset

Collecting malware dataset was the biggest part of our research. In this section, we have described the sources from where we collected all ransomware samples. We have referred many online sources in search of particular samples from selected ransomware families. We gathered 90% of our samples from Virus Total[5], and 8% were collected by automatically crawling into public malware repositories[6,7]. We captured the remaining by manually browsing through security forums. In order to confirm the malware sample to be a ransomware, we checked the MD5 hash values from Virus Total and then tagged it under a ransomware family when a majority of Antivirus engines recognized it from a particular family. In order to assign a sample a family name, we used Antivirus vendors' naming scheme. So, the naming policy was entirely based on the popularity of family name among Antivirus Engines. This paper covers the analysis of existing ransomware families observed from last few years. We have tried to cover the experimental analysis of 25 significant ransomware families. In order to have unbiased results, we performed analysis by taking at least three samples of each family.

3. Ransomware Analysis

In general, ransomware goes through various stages. Whenever an android device is infected with ransomware, then, first of all, it gains an administrative privilege by simply asking for it or by tricking the user by showing installing patch updates pop-up. Once it gains administrative access, it asks for permissions, which are required by an app in order to perform necessary tasks. But, we have seen from our analysis that, an app asking for irrelevant permissions in relative to the nature of that app is always for malicious purposes. Once, the permissions have been granted to the app, it starts gathering information from victim's device and then, it contacts command and control server. It sends this information to the attacker and these messages are usually encrypted with Transport Layer Security. It obtains a private key from command and control server in case of crypto ransomware. Using this key, it encrypts the selected files present in Android device. After completing encryption, it will ask the victim to pay the

ransom by showing an alert message. But in the case of locker ransomware, it will reset the PIN of the Android device and then asks for the ransom to restore access of the device.

In the case of Windows, we can see from Fig. 1 below that there are some main stages that every crypto family goes through. Each variant gets into victim's machine via any malicious website, email attachment or any malicious link and progress from there. Once the victim's machine gets infected, it contacts Command and Control server. It sends victim's machine information to the attacker and eventually obtains a randomly generated symmetric key from the server. Once it receives the encryption key, it then looks for specific files and folders to encrypt. Some variants look for all disk drives, network shares and removable drives as well for encrypting their data. Meanwhile, the malware deletes all the restore points, backup folders, and shadow volume copies.

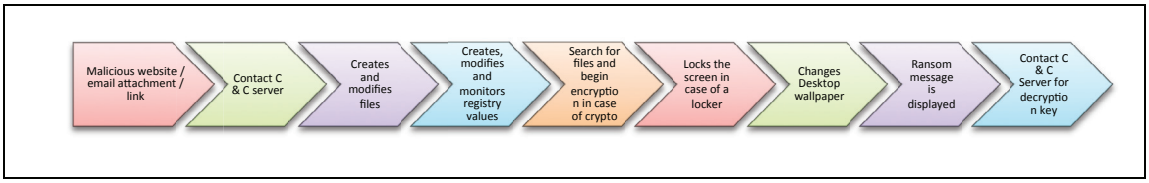

Fig. 1. Life cycle of Windows based ransomware

After the whole encryption process, it will display the ransom payment message on victim's machine. In the case of locker ransomware, malware goes through all the same phases but it doesn't do encryption of data. Once the victim's machine is infected with locker ransomware, it takes administrative privileges and takes control of the keyboard. It locks the user access to the machine. It changes the desktop wallpaper or it will show a window which notifies about ransomware attack and shows the steps to follow in order to get their access back. This paper uses static and dynamic malware analysis strategies in order to detect ransomware present in the device and the following subsections describe all the observations in detail.

3.1. Ransomware analysis on Android platform

Ransomware viruses exploit using the name of authorities including the FBI, USA Cyber Crime Investigation agency, and the ICE Cyber Crime Center. These locker type of ransomware make fake claims by portraying themselves as these agencies and then, warn the device users to pay some amount of money as fine for violation of law. Reverse Engineering process has been used to analyze ransomware Android malware. The ransomware analysis of the infected Android applications in this paper mainly focuses on AndroidManifest.xml and the source code of an application. Following are the malicious payloads which are observed during the analysis:

- **Privilege Escalation**: Once the whole application gets downloaded, then on opening the app, it asks for administration rights. Now, if the user clicks on activate button, then the application takes the privileges of device administrator, and this makes difficult to remove the malicious application from the device. In the recent versions of ransomware attacks, the activation window is overlaid with a malicious window pretending to be an Update patch installation. So, somehow the application tries to obtain the administrator privileges in order to lock the victim's device or to set a new PIN for lock screen of the device.
- **Remote control:** It was observed that earlier ransomware packages communicate with a website via HTTPS to get encryption keys. An application which tries to make a secure HTTP request to a suspicious target is a clear hint of malevolent purposes. Now, the new variants use XMPP communication in order to communicate with command and control server. These communications look like normal instant message communications, which makes the ransomware more difficult to get detected with anti-malware software. XMPP communications channel is used by the new Simplocker variants. Its variant uses an external Android library to communicate with the command and control network through a legitimate messaging relay server. And these messages can be encrypted using Transport Layer Security (TLS). The messages were received from the command and control network by the operators of the scheme via Tor. We observed that all communications to C&C server are done through port numbers 443, 80 and 123.

- **Information Collection:** We observed that ransomware applications collect information like IMEI number, call logs, contacts, profile, history bookmarks, SMS, the list of accounts in account service, phone state, GPS location of the phone, and IP address. Some of the ransomware even check the tasks running on the device. Simplocker family contacts Command and control server and sends the information found on the mobile device to the attacker.
- **Encryption Used:** Crypto ransomware like Simplocker and Pletor uses AES encryption scheme in order to encrypt the data present in SD card. It usually searches for the particular type of files and then encrypts them.
- **Permissions Used:** All apps which are installed by users requires some permissions to be granted to function properly. But malicious applications asks for permissions which are not for the functioning of the app for but the mischievous purposes. All the permission requests which don't seem to be in accordance with app services can be taken seriously and may not be granted. Table I below mentions all those permissions[8] which are used in general by benign level ransomware applications.

Table 1. Permissions used for benign level Ransomware.

Permission	Description
READ_PHONE_STATE	Allows read only access to phone state.
INTERNET	Allows applications to open network sockets.
ACCESS_NETWORK_STATE	Allows applications to access information about networks
WRITE_EXTERNAL_STORAGE	Allows an application to write to external storage.
READ_EXTERNAL_STORAGE	Allows an application to read from external storage.
RECEIVE_BOOT_COMPLETED	Allows an application to receive the ACTION_BOOT_COMPLETED that is broadcast after the system finishes booting.
ACCESS_COARSE_LOCATION	Allows an app to access approximate location.
ACCESS_WIFI_STATE	Allows applications to access information about Wi-Fi networks.
ACCESS_FINE_LOCATION	Allows an app to access precise location.
WAKE_LOCK	Allows using Power Manager Wake Locks to keep processor from sleeping.
INSTALL_SHORTCUT	Allows an application to install a shortcut in Launcher.
GET_TASKS	Allows an application to get information about the currently or recently running tasks.
CALL_PHONE	Allows an application to initiate a phone call without going through the Dialer user interface for the user to confirm the call.
CAMERA	Required to be able to access the camera device.

Highly malicious apps can ask for some more permissions which are being used by an attacker so that the removal of these applications gets more difficult. An app using permission like KILL_ BACKGROUND_PROCESSES in order to stop the antivirus processes running in order to prevent ransomware from detection. FACTORY_TEST is used to run an app as manufacturer test application using root user privileges. BIND_DEVICE_ADMIN ensures that only the system can interact with an application. These kind of permissions makes ransomware much more malicious and removal of these ransomware apps from Android devices becomes much more difficult.

3.2. Ransomware analysis on Windows platform

All variants were analyzed using Cuckoo Sandbox and Anubis. After analyzing, reports and traffic files were observed thoroughly and main observations were documented. We observed that ransomware component tries to get installed in the device by making some significant changes. These changes can be observed in File system activities, registry activities, and network communications. Following observations are from our experimental reports.

- **File System Activities:** Many files are read, modified, created and deleted during a ransomware's execution. A .txt file gets created at the starting of execution of ransomware and it gets modified constantly. Besides this sometimes .log, .tmp and .dmp file are also created and modified. All analyzed ransomware families modify PIPE\lsarpc file. PIPE\lsarpc interface generally communicates with the Local Security Authority subsystem. Our observation indicates that variants of Cryptowall family modify PIPE\lsarpc, MousePointManager and .exe file inside temp folder of the administrator account. Cryptowall variants also modify system.pif present under Start Menu so that it can restart selected programs even after rebooting of a machine. Wordpad.exe from CrypCTB family modifies ransomware.rtf and temp_cab_448359.cab in temp folder from Administrator locals folder. It also modifies MountPointManager, PIPE\lsarpc, PIPE\wkssvc, \Device\Afd\AsyncConnectHlp, \Device\Afd\Endpoint and \Device\RasAcd. PIPE\wkssvc interface manages the lanmanworkstation service. Acez.exe, which also belongs to CrypCTB family

seems to modify files such as Ip, \Device\Ip, and \Device\Tcp. Pjxbpxwfgnqu.exe from Cryptolocker family creates a set of .html, .png and .txt files inside C:\Documents and Settings\Administrator\ApplicationData\Adobe\Acrobat\8.0\Collab folder. It modifies Temporary Internet Files such as changing web browser's homepage and creating a text file as well which shows instructions to pay the ransom. It also modifies PIPE\lsarpc, PIPE\srvsvc, PIPE\wkssvc, \Device\Afd\Endpoint and \Device\RasAcd. Stub.exe from VaultCrypt family modifies Windows Update.exe, SysInfo.txt, MountpointManager, PIPE\lsarpc and PIPE\wkssvc. Midline.exe from Reveton family modifies C:\WINDOWS\system32\drivers\etc\hosts, MountPointManager, PIPE\lsarpc and \Device\Afd\Endpoint.

Most of the variants of these families use vssadmin tool to delete all volume shadow copies so that the victim can't recover the encrypted files. vssadmin.exe Delete Shadows/All/Quietcommand is used for deletion of files. Ransomware variants also try to delete restore points and all backup files. It also overwrites free disk space in order to prevent the files from being recovered.

- **Encryption Used:** All the recent variants of these families use a mixture of RSA and AES for encryption, RSA-2048 bit encryption as the public key and AES-256 bit encryption for encrypting data present in victim's machine. Almost all crypto ransomware families use the following approach. At first, crypto ransomware generates a randomly generated symmetric key, after this, it encrypts the selected files using this key with AES algorithm. After encrypting the data, it encrypts the randomly generated symmetric key with asymmetric public-private RSA key encryption algorithm. It is clear that the only the owner of private RSA key gets the randomly generated symmetric key. Recent variants use RSA 2048 bit encryption algorithm for generating an asymmetric public key and AES 256 bit key for file encryption. While the Virlock family is using 2 layers of encryption. The first layer is of XOR and ROL encryption and then, the second layer is of XOR encryption.
- **Registry Activities:** Most of the samples analyzed, showed that registry key is created whenever a ransomware program was installed. Ransomware variants modify many registry values. Here, we observed that many of the variants modify HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, HKU\S-1-5-21-842925246-1425521274-308236825-500\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders and changes AppData value to C:\Documents and Settings\Administrator\Application Data, cache value to C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files. Some variants modify HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Nls\Computer Name\ActiveComputerName and HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\WinLogon. Some keys like HKLM\System\CurrentControlSet\Control\Terminal Server checks if Terminal Server user is enabled or not. It makes sure that Language Hotkey and Layout Hotkey are also enabled. Ransomware reads the value of HKLM\Software\Microsoft\Cryptography\Defaults\Provider Types\Type 001, which is Microsoft Strong Cryptographic Provider. It also Checks the registry key CurrentVersion\Explorer\User Shell Folders for Application Data. We observed that variants of VaultCrypt, Cryptolocker, CrypCTB, Cryptowall, Reveton and Nymaim families were monitoring some Registry values. These registries keys are mentioned in Table II below.

Table 2. Monitored registry keys.

S. No.	Registry Value
1	HKLM\Software\Classes\CLSID
2	HKLM\Software\Microsoft\COM3
3	HKLM\Software\Microsoft\Tracing\RASAPI32
4	HKLM\System\CurrentControlSet\control\NetworkProvider\HwOrder
5	HKLM\System\CurrentControlSet\Services\WinSock2\Parameters\NameSpace_Catalog5
6	HKLM\System\CurrentControlSet\Services\WinSock2\Parameters\Protocol_Catalog9

Inside HKEY_LOCAL_MACHINE many registry values like CLSID, COM3, RASAPI32, HwOrder, NameSpace_Catalog5 and NameSpace_Catalog9 are monitored. CLSID is a 128-bit hexadecimal class identifier number, which identifies the component to the system. HKLM\System\CurrentControlSet\control\NetworkProvider\ tells about the default providers that gives access to SMB server, WebDAV server, and Microsoft RDP server.

- **Device control Communications:** We observed that Nymaim, Reveton, Vaultcrypt, Cryptlocker, and CrypCTB used some devices throughout the execution of ransomware and these devices are described in table III below:

Table 3. Devices used.

S. No.	Device	Description
1	\Device\KsecDD	Provides the kernel security device driver
2	\Device\Afd\Endpoint	Transfers packets to the local network and the Internet
3	\Device\Tcp	Used for TCP connections
4	MountPointManager	Driver responsible for maintaining persistent drive letters and names for volumes.

Rest of the ransomware variants used device \Device\KsecDD to communicate with victim's machine.

- **Network Activity:** Victim's machine tries to connect to a Command and Control server by initiating communication with a Client Hello using TLSV1. The server then responds back with a Server Hello. A certificate is sent from the server to victim's machine. Once the communication between the two has been established then, victim machine starts Client key exchange, Change Cipher Spec, and Encrypted Handshake message to get the encryption keys and other messages. We used Wireshark to capture the traffic and observed that all TCP connections were made on port 80 and 443 and UDP connections were made on port 53. Lastly, an Encrypted Alert message is sent from victim machine to C & C server.
- **Locking mechanism:** Locker ransomware variants use Javascript code to change browser settings of browsers like Chrome, Firefox, Internet Explorer, Safari etc. The code creates an iframe loop which loads the fake message claiming to be coming from the local Police Agency. The message includes a ransom note and displays the procedure of making payment as well. Reveton variants read many registry values and modify them. From the registry values which are being read, modified, deleted and monitored, we can say the how the locker interacts with victim's machine in order to perform its tasks. Reveton variants are stealing information as well, which we captured from our analysis results that many keystrokes get captured during execution.

4. Ransomware Evolution

Ransomware came into existence a long time ago. As we can see in Figure 2, the first Windows ransomware started to spread in 1989 and since then it has been present till now but has changed significantly since then. The first ransomware attack was PC Cyborg attack, which was seen in December 1989. It was the first crypto type ransomware as it used the combination of a symmetric key and an Initialization vector to encrypt the files present in the computer drives.

We have seen three types of locker ransomware till now – SMS ransomware, MBR ransomware, and Fake FBI ransomware. The first Fake Antivirus ransomware appeared in 2004 and then in 2005 we saw a series of Fake Antivirus ransomware types. Some of these were named as Spysherriff, Performance Optimizer, and Registry care.

In 2005, the PGPcoder family started evolving and this clearly indicates the era of crypto ransomware. Gpcode used custom encryption scheme for encryption of data. PGPcoder spread wildly till 2008 as we can see many variants. In 2006, two more families started spreading, these are Cryzip and Archiveus. Cryzip searched for files with selected extensions, and then placed these encrypted files in a zipped folder. Archiveus placed all the files in a password protected folder.

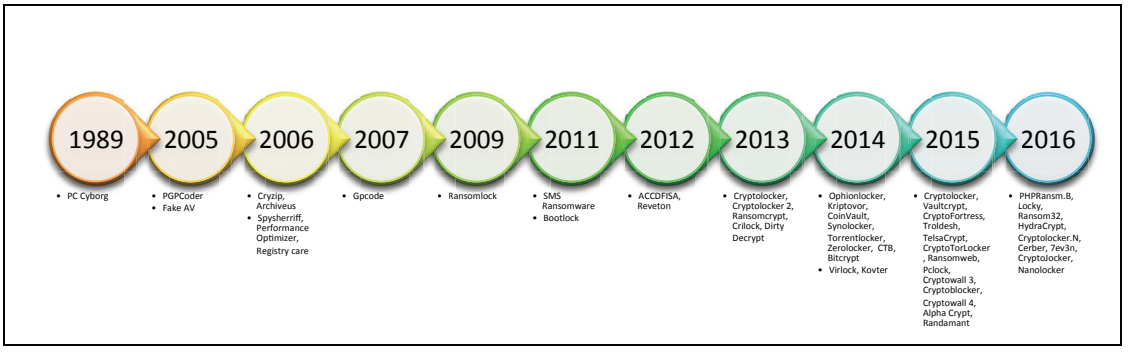

Fig. 2. timeline for Windows based ransomware

MBR Ransomware came into existence in 2010, the first variant that we came across was Trojan-Ransom.Boot.Seftad.a, and in 2011 bootlock.B arrived. This type of ransomware replaces the original MBR with its own code and then locks the user from accessing its services. It never encrypts file and displays the ransom message at computer boot-up time.

Fake AV was spread in the wild in 2004. It became significant in 2005 when it tried to take the form of Fake Antivirus solutions, Performance Optimizer software and Registry care software, which tried to offer paid solutions for your machine problems which didn't even existed. It was surfaced over the internet till 2008.

Fake FBI ransomware arrived in 2011 with the Ransomlock family. Later in 2012, families like Reveton and ACCDFISA started spreading in-the-wild. These families display the fine payment notice from official looking local law enforcement agencies. Later, many variants of Ransomlock and Reveton came in 2013. In 2014, new locker families like Virlock, Kovter and few new variants of Ransomlock arrived.

Crypto ransomware became a huge problem in 2013 when it did a comeback with Cryptolocker, Cryptolocker 2, Ransomcrypt, Crilock and Dirty Decrypt. Later in 2015, we saw new variants of Ransomcrypt,Cryptolocker, Vaultcrypt, CryptoFortress, Troldesh, TelsaCrypt, CryptoTorLocker, Ransomweb, Pclock, Cryptowall 3, Cryptoblocker and Cryptowall 4. Cryptowall 3 uses Tor anonymity network for C & C communication. Almost all recent crypto ransomware families are using very sophisticated encryption techniques. Recently in 2016, new families of crypto like PHPRansm.B, Locky, Ransom32, HydraCrypt, Cryptolocker.N andCerber have started to spread.

In the case of Android, we observed that the first locker ransomware appeared in 2012, which was detected by antivirus engines as Genric.17.1762. Later in 2013, LockDroid, Kovter, Sypeng and Pletor arrived. In 2014 Simplocker, Koler, ScarePackage, ScareMeNot, ColdBrother, Jisut, Locker, LockerMaster and many variants of LockDroid came into existence. In 2015, Lockerpin, FakeInst, SMSSend, Agent, HiddenApp, Slocker families started surfacing all over the Android devices. Recently in 2016, we haven't seen any new family but have noticed new variants of lockDroid, locker, and slocker families. Figure 3 shown below shows the timeline for Android based ransomware which states how ransomware started to surface all over Android devices since 2012.

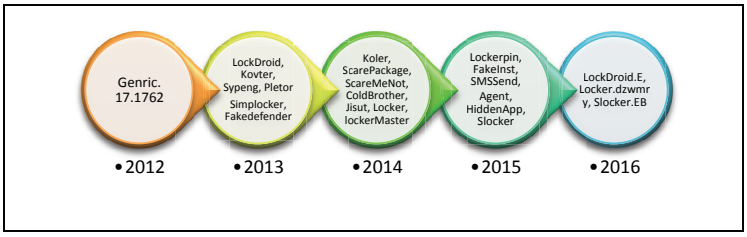

Fig. 3. timeline for Android based ransomware

Fakedefender was the first Fake AV seen in 2013. This Fake AV emphasize on purchasing Antivirus solutions to remove malware from your device, and these malware were not even present in the device. The first crypto android ransomware Simplocker came in 2013. It uses AES Encryption to encrypt files.

Crypto ransomware like Simplocker and Pletor encrypts files present on Android device's memory card. Simplocker family malware scans the SD card and searches for files with jpeg, jpg, png, bmp, gif, pdf, doc, docx, txt, mp4, mkv, avi and then uses AES algorithm to encrypt these file's data.

5. Ransomware Detection

For detection of ransomware in Windows operating systems, we checked the MD5 checksum values for each analyzed sample and checked it against Antivirus search engines by using File Fingerprinting technique. PEiD[9] tool used to detect packers, cryptors and compilers found in PE files. This helped in finding out whether an executable is malicious or not even before installing an application. PEView[10] is another tool which gives basic information regarding PE. RegShot is capable of taking snapshots of all registry values before and after the installation of a

program and later they can be compared. For better detection results, continuous monitoring of registry activities should be done.

For detection of ransomware in Android environment, we have used Apktool[11] to extract AndroidManifest.xml and source code from the application. An algorithm can be developed to detect AndroidManifest.xml file before the installation of an application. Permissions should be checked and denied which looks suspicious in an application. Giving the user an option to grant only necessary permissions to an application at the time of installation can solve many problems.

6. Conclusion and Discussion

In this paper, we performed analysis of ransomware families, mainly focusing on their evolution and characterization. The characterization of ransomware families is based on ransomware samples from 25 families that have emerged over the last few years. Our results show that a significant number of ransomware families exhibits very similar characteristics. Section III reveals how a ransomware interacts with the file system, registry activities, and network operations when a machine is under a ransomware attack. In case of the Windows, implementing practical defense mechanisms is possible, by continuously monitoring the file system activity and registry activity. We have seen above that a particular set of registry keys are being monitored and modified, so if these registry values are put under continuous observation then, detection of ransomware is possible. We also observed that Windows 10 is quite effective against ransomware. To check the effectiveness of Windows 10, we tried to infect a machine with ransomware variants. This machine already had all inbuilt security procedures upgraded and running, then Windows 10 automatically detected and deleted all those ransomware variants. In the case of Android, as we have seen above that there are also few things by which we can detect ransomware by paying close attention to the AndroidManifest file and the permissions required by an app.

Best practices to prevent the user's data from getting into unrecoverable state, a user should have an incremental online and offline backups of all the important data and images. Moreover, all the inbuilt defense mechanisms and above mentioned detection tools should be kept up and running all the time. Ransomware has started to spread on various platforms now. Recently, it has started to attack Linux and Mac operating systems[12]. So, analysis of ransomware can be done on these platforms for future research work.

References

1. Wikipedia, Bitcoin, [Online]. Available: https://en.wikipedia.org/wiki/Bitcoin
2. CIO, Ransomware_InfoSheet, [Online]. Available: http://www.cio.gov.bc.ca/local/cio/informationsecurity/pdf/Ra nsomware_InfoSheet.pdf.
3. Anubis and Andrubis, http://anubis.iseclab.org/.
4. Cuckoo Sandbox, https://www.cuckoosandbox.org/.
5. Virus Total - Intelligence Search Engine, https://www.virustotal.com.
6. AvCaesar, https://avcaesar.malware.lu/.
7. Contagio, [Online]. Available: http://contagiodump.blogspot.ca/2010/11/links-and-resources-for-malware-samples.html.
8. Developer.android, Manifest Permissions, [Online]. Available: https://developer.android.com/reference/android/Manifest.permission.html
9. PEiD, http://www.forensicswiki.org/wiki/PEiD
10. PEView, https://www.aldeid.com/wiki/PEView
11. Apktool, http://ibotpeaches.github.io/Apktool/
12. The Register, John Leyden, First OS X ransomware actually a scrambled Linux file scrambler [Online]. Available: http://www.theregister.co.uk/2016/03/09/first_macosx_ransomware_actually_linux_port/