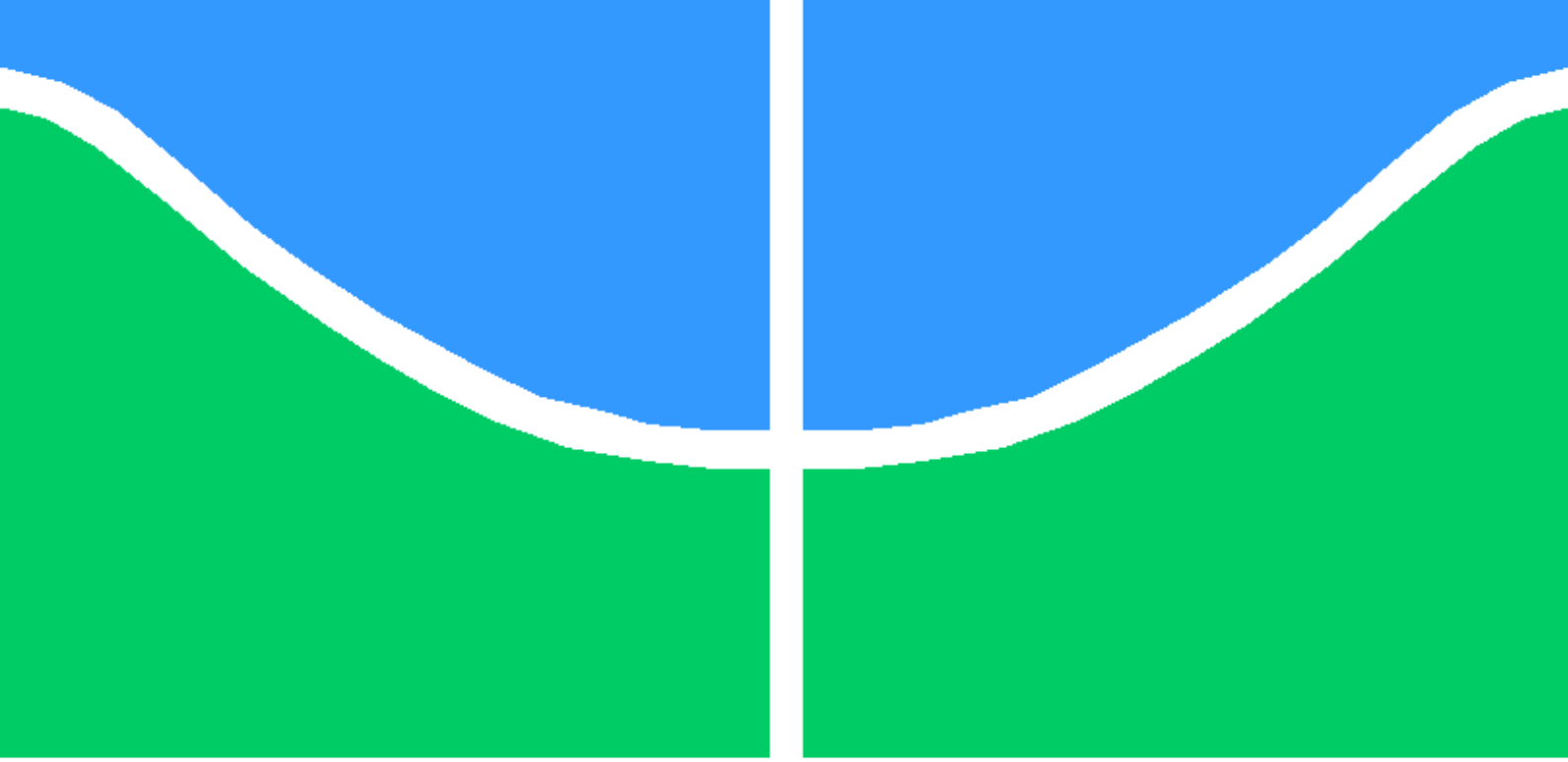

TRABALHO DE GRADUAÇÃO

ANÁLISE E IDENTIFICAÇÃO DE PERFIS DE OPERAÇÃO DE RANSOMWARE

Douglas Fernandes Barbeta

Gustavo Brito Flores

Brasília, Novembro de 2017

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

ANÁLISE E IDENTIFICAÇÃO DE PERFIS DE OPERAÇÃO DE RANSOMWARE

Douglas Fernandes Barbeta

Gustavo Brito Flores

Relatório submetido ao Departamento de Engenharia Elétrica, como requisito parcial para obtenção do grau de Engenheiro de Redes de Comunicação

Banca Examinadora

Prof. Flávio Elias Gomes de Deus, ENE/UnB _____
Orientador

Prof. Robson de Oliveira Albuquerque, ENE/UnB _____
Coorientador

Prof. Dr. Rafael Timóteo de Sousa Jr, ENE/UnB _____
Examinador Interno

Dedicatórias

Primeiramente a Deus, aos meus familiares que sempre me apoiaram nas minhas decisões e aos meus professores pelos ensinamentos.

Gustavo Brito Flores

Dedico este trabalho a Deus e aos meus familiares que estão ao meu lado sempre.

Douglas Fernandes Barbeta

Agradecimentos

Agradeço primeiramente a Deus, por todas as bênçãos durante minha vida. Agradeço à minha família, por todo o apoio dado nessa jornada de estudos, desde o colégio até a universidade, em especial ao meu irmão Danilo e meus pais Décio e Celma. Agradeço aos meus amigos sempre estiveram ao meu lado, que sempre me incentivaram e estiveram presentes nos momentos tristes e alegres. Aos meus professores da graduação, especialmente ao orientador Flávio Elias e coorientador Robson Albuquerque, que transmitiram os conhecimentos essenciais a respeito de nosso trabalho com muita paciência e dedicação.

Douglas Fernandes Barbeta

Gostaria de agradecer primeiramente a Deus. Aos meus pais, Antônio e Ivanete e meus irmãos Mateus e Daniela, por me aconselharem e sempre me apoiarem em qualquer decisão. Agradeço a todos meus professores durante minha graduação, em especial, ao Flávio Elias e Robson Albuquerque, pelos ensinamentos e total atenção em nosso trabalho. Agradeço também aos meus amigos, sem eles nada disso teria sido possível.

Gustavo Brito Flores

RESUMO

Com o desenvolvimento de novas tecnologias, o mundo está cada vez mais globalizado, a internet, a qual é um sistema global de redes de computadores, está cada vez mais presente e indispensável na era da informação. Paralelamente a esse avanço, usuários com conhecimento mais avançado e malicioso perceberam uma oportunidade de se beneficiar com tal cenário, desenvolvendo *softwares* malígnos com o objetivo de causar algum dano, alterações ou roubo de informações. Esse tipo de *software* se caracteriza como um *malware*. Uma vez que cada tipo de *malware* tem um comportamento específico, o *Ransomware*, classificado como um *malware*, possui como principais características o bloqueio de serviços e criptografia de dados.

Analisando essa situação, este trabalho foi desenvolvido com o objetivo de analisar os aspectos de diversas famílias de *Ransomwares*, no intuito de obter resultados que caracterizem o comportamento desse tipo de *malware* em específico.

Por meio da ferramenta *Cuckoo Sandbox* foi possível analisar os *Ransomwares* coletados nas bibliotecas de dados *Malware Traffic Analysis*, *the Zoo* e *Open Malware*, sendo que no final do processamento do arquivo malicioso, o *Cuckoo Sandbox* gera um arquivo HTML sendo aberto no navegador da máquina do usuário contendo um relatório completo das atividades e caraterísticas do malware analisado. A etapa de envio do arquivo malicioso e retorno do relatório foi implementado em um script na linguagem *Shell Cript*, tornando toda essa execução, um processo completamente automatizado.

ABSTRACT

With the development of new technologies, with the world increasingly globalized, the internet, which is a global system of computer networks, is increasingly present and indispensable in the information age. Parallel to this advance, users with slightly more advanced and malicious knowledge perceived an opportunity to benefit from such a scenario, developing malicious software with the objective of causing some damage, alterations or theft of information. This type of software is characterized as a malware. Since each type of malware has a specific behavior, Ransomware, classified as malware, has as main characteristics the blocking of services and encryption of data.

Analyzing this situation, this project was developed with the objective of analyzing the aspects of the several families of textit Ransomwares, in order to obtain results that characterize the behavior of this type of textit malware in specific.

Through the Cuckoo Sandbox tool it was possible to analyze the Ransomwares collected in the data libraries Malware Traffic Analysis, the Zoo and Open Malware, in the end of the malicious file processing, Cuckoo Sandbox generates an HTML file being opened in the browser of the user's machine containing a complete report of the activities and characteristics of the malware analyzed. The step of sending the malicious file and returning the report was implemented in a script in Shell Cript language, making all this execution a completely automated process.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

Acrônimos

AES	Advanced Encryption Standard
API	Application Program Interface
APK	Android Package
ARP	Address Resolution Protocol
C&C	Comando e Controle
DDoS	Distributed Denial of Service
DGA	Domain Generation Algorithm
DLL	Dynamic-link library
DNS	Domain Name System
FBI	Federal Bureau of Investigation
ICMP	Internet Control Message Protocol
GNU	GNU's Not Unix
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
iOS	iPhone Operating System
IP	Internet Protocol
IRC	Internet Relay Chat
JPG	Joint Photographics Experts Group
JSON	JavaScript Object Notation
MP3	MPEG Layer 3
MBR	Master Boot Record
MD5	Message Digest Algorithm 5
NAS	Network-Attached Storage
NTFS	New Technology File System
OCR	Optical Character Recognition
P2P	Peer-to-Peer
PDF	Portable Document Format
PE32	Portable Executable 32-bit
PNG	Portable Network Graphics
RSA	Rivest Shamir Adleman
SHA	Secure Hash Algorithm

Acrônimos

SMTP	Simple Mail Transfer Protocol
SSL	Secure Socket Layer
SO	Sistema Operacional
SP3	Service Pack 3
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
VM	Virtual Machine
WPA	Wi-Fi Protected Access
WPA2	Wi-Fi Protected Access II

Capítulo 1

Introdução

Com o avanço da tecnologia da informação, os métodos de segurança desta se tornam cada vez mais presentes no cotidiano dos usuários, devido ao fato de que os ataques cibernéticos aumentam a cada ano e possuem maiores e melhores ferramentas para torná-los discretos e também devastadores. [Digital , 2017]

O primeiro caso de infecção por *Ransomware* foi registrado em 1989 durante a conferência internacional da Organização Mundial de Saúde sobre a AIDS *(Acquired Immune Deficiency syndrome)*, o que deu a esta versão a alcunha de "AIDS Trojan" [Francis, 2016]. A primeira versão deste *malware* foi distribuída por meio de disquetes. As versões que seguiram tinham foco na pasta de documentos dos usuários do sistema infectado.Os métodos empregados evoluíram ao longo do tempo, acompanhando a tecnologia. Os métodos de distribuição passaram de disquetes para anexos de *e-mail*, páginas da *web* comprometidas, falhas em *plug-ins* e outros aplicativos, outras infecções por *software* mal-intencionado, e campos de propaganda em páginas da web.

Neste trabalho foram realizados testes e análises em diversos *Ransomwares* de diferentes famílias, com o intuito de identificar os seus perfis de operação e as técnicas de ataque utilizadas por cada um. O *Cuckoo Sandbox* foi o software utilizado para realizar tais análises, sendo objeto alvo uma máquina virtual *VirtualBox* com o sistema operacional Windows XP SP3. Para submeter os *malwares* à esta análise, foi desenvolvido um script que automatiza o envio do *malware* para a máquina virtual, executa o mesmo e por fim, concluindo esta execução, um relatório HTML é gerado e apresentado para o usuário com as características e peculiaridades de cada *malware* durante o processo de ataque.

1.1 Motivação

Os maiores problemas enfrentados são relacionados à variedade de tipos de *Ransomware*, sua forma específica de atuar e o grande avanço de tecnologias. Os usuários, domésticos ou empresariais, ainda possuem dificuldades em saber se prevenir de um ataque de *Ransomware*, visto que é um *malware* relativamente recente e com um *modus operandi* bastante específico comparado a outros *malwares*.

Analisando a figura 1.1, pode-se observar a variedade de famílias ou tipos diferentes deste *malware*. O gráfico retirado da *MalwareBytes Labs* apresentado na figura demonstra o percentual de ataques de algumas famílias de *Ransomware* de acordo com os meses do ano de 2016.

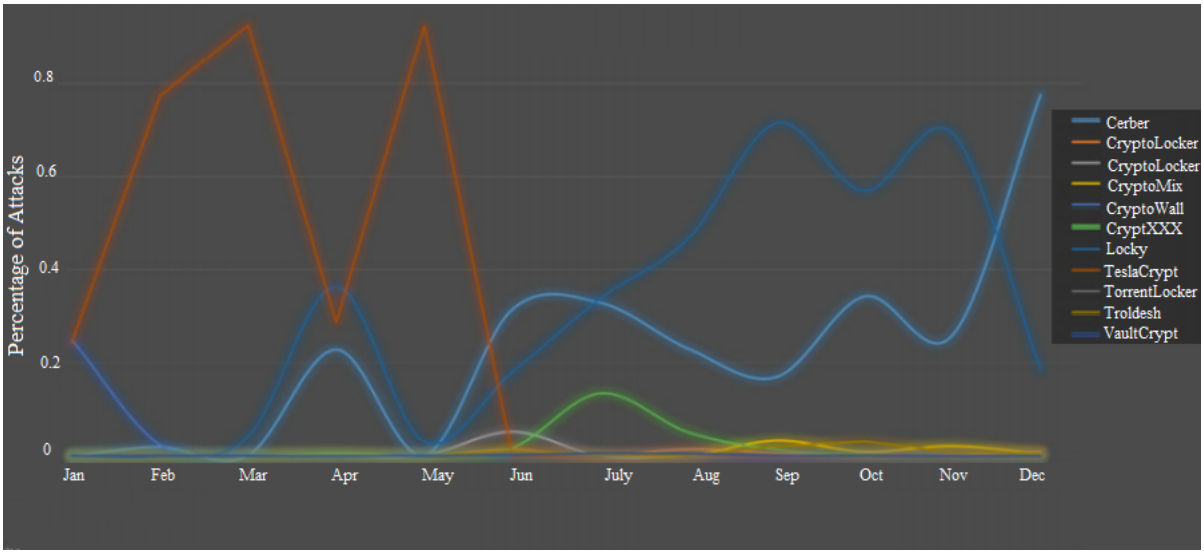

Figura 1.1: Percentual de ataques por família de *Ransomware* durante o ano de 2016 [Malwarebytes , 2017]

Além disso, ainda há dificuldade de detectá-lo no exato momento em que está acontecendo o ataque. Isso é referente à quantidade de atualizações e sofisticações que os *hackers* adicionam neste tipo de *malware* e que os mecanismos de defesa dos sistemas operacionais, anti-vírus e programas anti-*Ransomware* conseguem detectar, porém as mais novas famílias desse *malware* ainda não são totalmente detectadas. Nas figuras 1.2 e 1.3 é possível notar a evolução do *Ransomware* referente ao aumento deste tipo de ataque em relação à Janeiro de 2016 e Novembro de 2016:

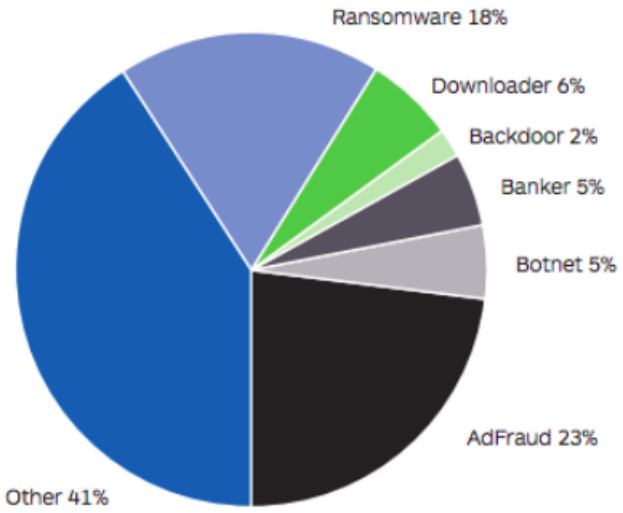

Figura 1.2: Percentual dos tipos de ataques cibernéticos em Janeiro de 2016 [Malwarebytes , 2017]

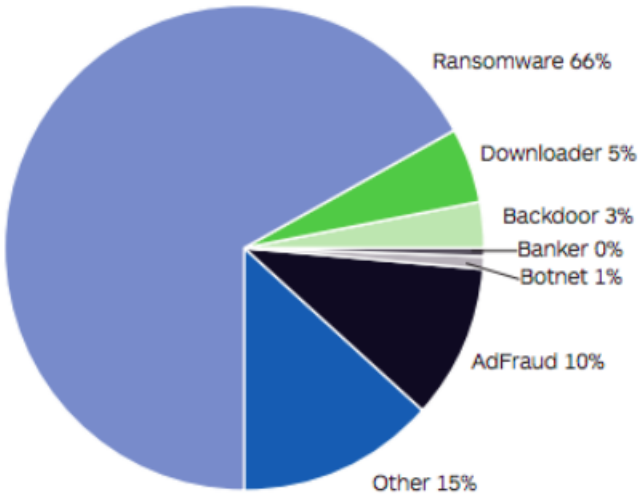

Figura 1.3: Percentual dos tipos de ataques cibernéticos em Novembro de 2016 [Malwarebytes , 2017]

De forma mais atualizada, a figura 1.4 mostra ainda mais o quanto as variações deste tipo de *malware* aumentaram entre 2016 e 2017.

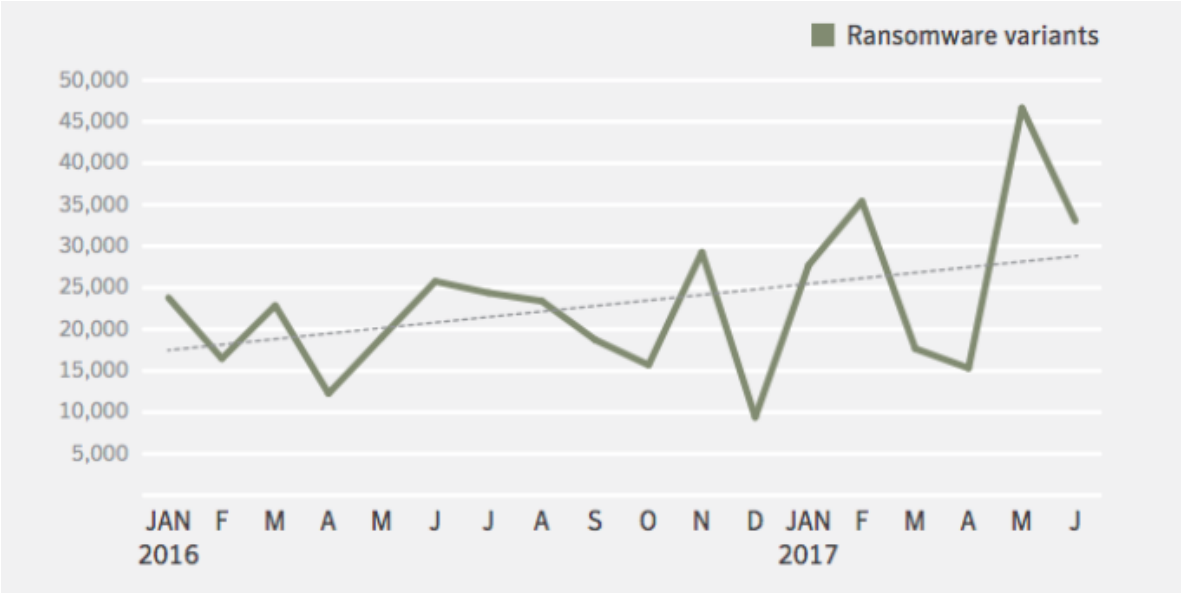

Figura 1.4: Novas variações de *Ransomware* por mês desde o ano de 2016 até o ano de 2017 [Symantec , 2017]

Com o seu crescente desenvolvimento, o *Ransomware* atingiu o patamar de malware mais rentável da história. No mês de Julho deste ano, observado na figura 1.5, o *Ransomware* da família *NotPetya* atacou empresas nos EUA e na Europa. Um dos ataques de maiores impactos foi na empresa *AP Moller-Maersk*, que movimenta cerca de um quinto do frete mundial.[Forbes , 2017]

De acordo com um comunicado emitido pela empresa, o custo total para lidar com os prejuízos foi na faixa de US $ 200 a US $ 300 milhões. Os analistas da indústria esperam que as empresas de todo o mundo percam um total de $ 5 bilhões envolvendo *Ransomwares* neste ano de 2017. Em 2015, o FBI encerrou as perdas totais em um valor de aproximadamente US $ 1,7 bilhão [Lee Mathews, 2017].

Observado todo esse impacto, foi interessante estudar os aspectos e comportamentos deste software malicioso, com o intuito de evitar ou minimizar futuros ataques.

Figura 1.5: Ataque do *Ransomware NotPetya* à companhia Maersk [CNBC , 2017]

1.2 Objetivos

Este trabalho tem como objetivo realizar a análise do *malware* do tipo *Ransomware*, identificá-lo e compará-lo em relação à sua atividade primária e seu comportamento, levando em consideração alguns aspectos:

- Realização de análise automatizada de um *Ransomware* por meio de uma solução de script desenvolvida;
- Caracterização da atividade de um *Ransomware*;
- Vulnerabilidades exploradas por este *malware*;

- Definição de sua técnica de ataque baseando-se nos processos gerados no S.O. a partir de um relatório gerado automaticamente após a execução do *malware* na máquina alvo.

1.3 Estrutura do Trabalho

No Capítulo 2, na Fundamentação Teórica, serão abordados os principais fundamentos que foram utilizados nesse trabalho como os conceitos básicos de um *Ransomware*, as vulnerabilidades exploradas e as técnicas de ataque utilizadas por este.

No Capítulo 3, serão abordados a arquitetura desenvolvida e as ferramentas utilizadas neste trabalho, com suas principais características. Cada seção descrita será responsável por cada *software* que foi utilizado.

No Capítulo 4, serão feitas análises em diversas amostras de famílias de *Ransomware* com as ferramentas apresentadas no Capítulo 3, além de apresentar comparações a partir dos resultados que foram obtidos.

Por último, o Capítulo 5, contém as conclusões e limitações acerca das análises feitas, e possíveis trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Neste capítulo será apresentado os conceitos, características e fundamentos para o entendimento do trabalho. É feita a apresentação do conceito de *malware*, suas características e classificações. Por sua vez, será bem detalhado a definição de *Ransomware*, sua classificação, e divisão de suas diversas famílias. Por fim, é abordado suas formas de ataques a computadores pessoais, dispositivos móveis e servidores.

2.1 *Software* Malicioso *(Malware)*

Um *malware* é um *software* malicioso destinado a infiltrar-se em um sistema de computador alheio de forma ilícita, com o intuito de causar alguns danos, alterações ou roubo de informações. Ele pode aparecer na forma de código executável, scripts de conteúdo ativo, e outros *softwares*. O termo *malware* só se aplica a *software* que intencionalmente cause danos. Alguns dos principais tipos de *malware*, além do próprio *Ransomware* são:

Tabela 2.1: Tipo de *malware* e sua descrição

Malware	Descrição
Vírus	Propaga-se infectando cópias de si mesmo e se tornando parte de outros programas e arquivos de um computador. O vírus depende da execução dos arquivos hospedeiros para que possa se tornar ativo e continuar o processo de infecção.
Worm	Capaz de se propagar automaticamente através de redes, enviando cópias de si mesmo de computador para computador. Diferente do vírus, o *worm* não embute cópias de si mesmo em outros programas ou arquivos, e não necessita ser executado para se propagar. A sua propagação dá-se através da exploração de vulnerabilidades existentes ou falhas na configuração de *softwares* instalados em computadores.

Trojan	É um tipo programa malicioso que pode entrar em um computador disfarçado como um programa comum e legítimo. Ele serve para possibilitar a abertura de uma porta de forma que usuários mal intencionados possam invadir a máquina.
Keylogger	Captura e armazena as teclas digitadas pelo usuário no teclado do computador. Normalmente, a ativação é condicionada a uma ação prévia do usuário.
Screenlogger	Forma avançada de *keylogger*, capaz de armazenar a posição do cursor e a tela apresentada no monitor, nos momentos em que o *mouse* é clicado.
Spyware	Tem objetivo de monitorar atividades de um sistema e enviar as informações a terceiros. Podem ser usados de forma legítima, mas geralmente, são usados de forma dissimulada, não autorizada e maliciosa.
Adware	Projetado para apresentar propagandas. É comum aparecerem na hora de instalar um programa.
Backdoor	Permite a um invasor retornar a um computador comprometido. Normalmente, este programa é colocado de forma a não ser notado.
Exploits	Projetado para explorar uma vulnerabilidade existente em um *software* de computador.
Sniffers	Usado para capturar e armazenar dados trafegando em uma rede de computadores. Pode ser usado por um invasor para capturar informações sensíveis (como senhas de usuários), em casos onde esteja sendo utilizadas conexões sem criptografia. Deixa a placa de rede em modo promíscuo.
Port Scanners	Para efetuar varreduras em redes de computadores, com o intuito de identificar quais computadores estão ativos e quais serviços estão sendo disponibilizados por eles. Amplamente usados por atacantes para identificar potenciais alvos, pois permite associar possíveis vulnerabilidades aos serviços habilitados em um computador.
Bot	Além de incluir funcionalidades de *worms*, dispõe de mecanismos de comunicação com o invasor, permitindo que o programa seja controlado remotamente. O invasor, ao se comunicar com o bot, pode orientá-lo a desferir ataques contra outros computadores, furtar dados, enviar spam, etc.
Root Kit	Conjunto de programas com o fim de esconder e assegurar a presença de um invasor em um computador comprometido.
Quantum	Cria site falso para implantar sistemas.

2.2 *Ransomware*

Ransomware, por sua vez, é um *malware*, que precisa de outro mecanismo, geralmente arquivos ou brechas de segurança, para infectar um hospedeiro. Os mecanismos evoluíram desde então e têm causados muitos prejuízos, não só para usuários comuns como para órgãos governamentais e

empresas de todo o mundo.[CERT.br , 2015]

O objetivo principal de um *Ransomware* é raptar arquivos ou o próprio sistema do hospedeiro infectado. O rapto de arquivos é feito com criptografia forte e o rapto do sistema geralmente utiliza *rootkits* que bloqueiam o funcionamento normal do sistema operacional e/ou aplicativos. É solicitado um resgate (do inglês *"ransom"*) em dinheiro ou moeda digital (*bitcoin*) para restaurar os arquivos ou o sistema, o que torna este tipo de ataque bastante lucrativo.[Infowester , 2016]

A primeira versão deste vírus foi distribuída em 1989. As versões que seguiram tinham foco na pasta de documentos dos usuários do sistema infectado. O rapto de sistemas surgiu por volta de 2009 e exigiam o pagamento em nome da fabricante do sistema operacional, alegando uso indevido ou não licenciado; ou em nome de agências governamentais de inteligência sob o pretexto de acesso a material indevido na internet. Em alguns casos, a *Master Boot Record* (MBR) do disco era substituída para que nada mais pudesse ser executado no hospedeiro, senão o *software* mal-intencionado [Savage; Coogan; Lau, 2015].

Os métodos empregados evoluíram ao longo do tempo, acompanhando a tecnologia. Os métodos de distribuição passaram de disquetes para anexos de *e-mail*, páginas da *web* comprometidas, falhas e brechas em plugins e outros aplicativos, outras infecções por *software* mal-intencionado, e campos de propaganda em páginas da *web*. As páginas comprometidas deixaram de ser exclusividade de domínios ditos "suspeitos" como os que ofereciam pornografia e se espalharam por toda a Internet, principalmente por meio de *plug-ins* e propagandas. As famílias mais populares deste malware atualmente estão focadas em rapto de arquivos, em qualquer parte do sistema. Em alguns casos, a chave utilizada na criptografia é compartilhada com um servidor central de comando e controle. Se a criptografia for assimétrica como o RSA, a chave privada existe somente no servidor. Se for simétrica, a chave é enviada ao servidor utilizando conexão segura semelhante a SSL/TLS. Essa troca de chave é feita para eventual recuperação dos arquivos raptados, embora nem todos os ataques efetivamente são recuperados os arquivos [Savage; Coogan; Lau, 2015].

O *Ransomware* se tornou cada vez mais complexo. Os codificadores utilizam técnicas de criptografia RSA cada vez mais sofisticadas e códigos cada vez mais longos. Em meados de 2006, o *Ransomware Gpcode.AG* utilizava uma chave RSA pública de 660 bits RSA. Em dois anos, a sua nova variação utilizava uma chave de 1024 bits, o que era quase impossível de decifrar. Na atualidade, um subgrupo de *Ransomware CryptoLocker* utiliza uma chave RSA de 2048 bits.[Livehacking , 2010] Além disso, a forma de pagamento também evoluiu com o tempo, passando de depósitos e transferências diretas para cartões pré-pagos, vales e mais recentemente *BitCoins*, que é uma moeda eletrônica anônima por definição. De acordo com um relatório da Cisco, ele domina o mercado de *malware* e é o tipo de *malware* mais rentável da história. [Savage; Coogan; Lau, 2015].

Existem dois principais tipos de *Ransomware*: *Locker* e *Crypto*. Dentro desses principais tipos, existem subdivisões em famílias, que são agrupados de acordo com sua característica específica de ataque. [Kaspersky , 2015]

2.2.1 *Locker*

É um tipo de *Ransomware* que foi projetado para negar/bloquear o acesso do usuário a recursos do computador ou equipamento. Geralmente assume a forma de bloqueio da interface do usuário do computador ou do dispositivo e, em seguida, pede ao usuário que pague uma certa taxa, dependendo da família do *Ransomware*, para restaurar o acesso a ela. [Computer , 2017]

Os computadores uma vez bloqueados, possuem capacidades de acesso e controle muito limitados, como somente permitindo ao usuário interagir com o *Ransomware* e pagando o resgate. Isso significa que, por exemplo, o acesso ao *mouse* pode estar desabilitado e no caso do teclado a funcionalidade pode ser limitada a chaves numéricas, permitindo que a vítima digite apenas números para efetuar o pagamento do resgate. Na figura 2.1 é possível observar um exemplo de solicitação de resgate do *Locker Ransomware*.

Figura 2.1: Exemplo de solicitação de resgate do *Reveton - Locker Ransomware* [KrebsOn , 2012]

2.2.2 *Crypto*

É um tipo de *Ransomware* que cifra arquivos e dados armazenados no equipamento de um usuário. A criptografia codifica o conteúdo de um arquivo, de modo que é ilegível pelo usuário ou pelo próprio dispositivo. Para restaurá-lo para uso normal, é necessária uma chave de decifragem.[F-Secure , 2016]

Quando o *Crypto Ransomware* criptografa os arquivos de um usuário, é essencialmente tornar

esses arquivos “reféns”, em seguida, uma demanda de resgate é exibida oferecendo ao usuário a chave de decifragem necessária para restaurar os arquivos, se uma quantia especificada for paga. Em alguns casos, o usuário possui um período de tempo limitado para efetuar o pagamento. Na figura 2.2 é apresentado um exemplo de solicitação de resgate do *Crypto Ransomware*.

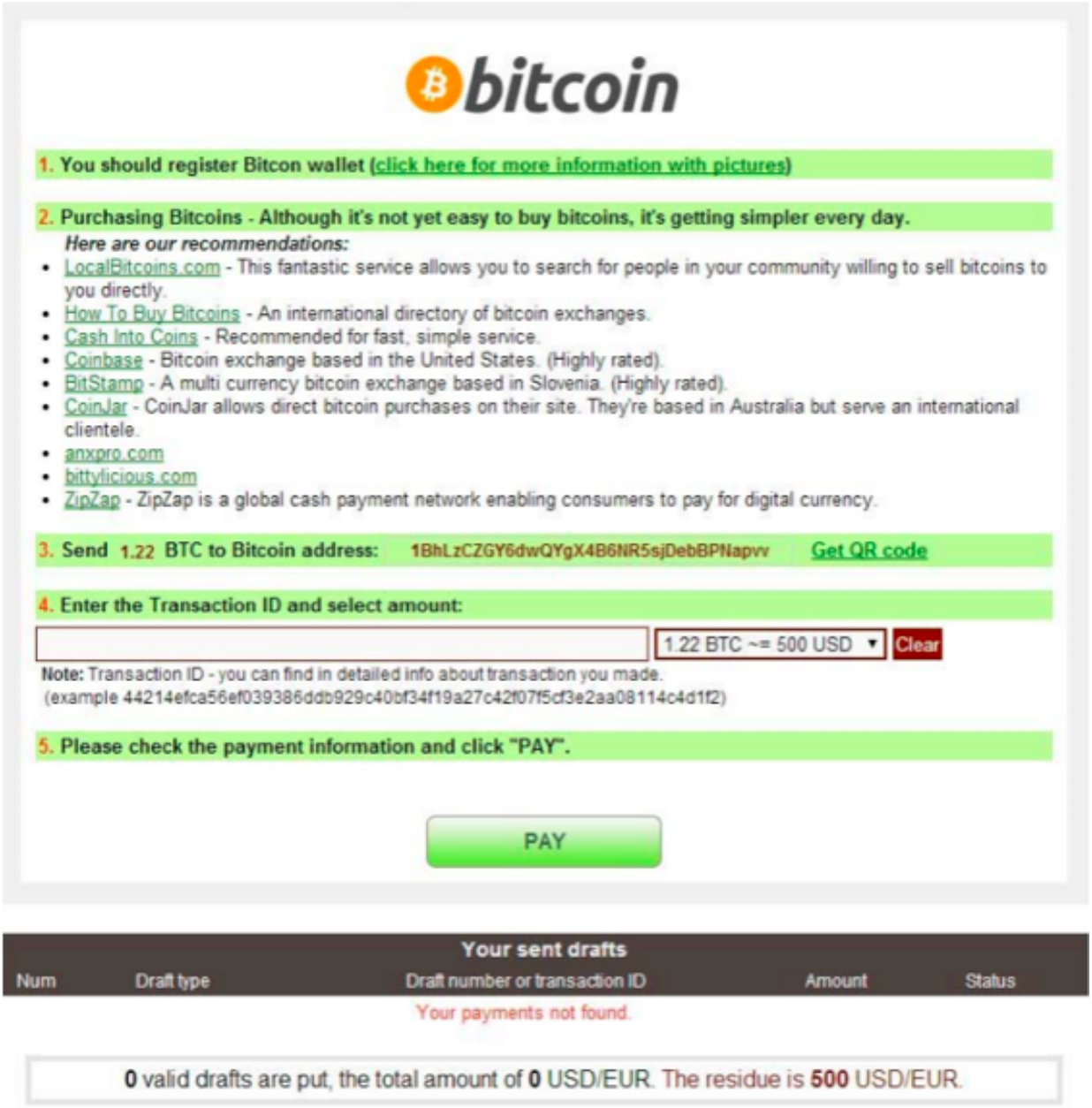

Figura 2.2: Exemplo de solicitação de resgate do *CryptoWall - Crypto Ransomware* [Kafeine , 2016]

2.2.3 Famílias de *Ransomware*

Na tabela 2.2 estão apresentados as principais famílias de *Ransomware*, o tipo em que ele é classificado e sua descrição com características e processos de ataque.[TrendMicro , 2017]

Tabela 2.2: Principais famílias de *Ransomware* e descrição

Família	Tipo	Descrição
ACCDFISA	Crypto	Descoberto no início de 2012, criptografa arquivos com uma senha. Os cibercriminosos por trás desse Ransomware pedem o pagamento através do *Moneypak*, *Paysafe* ou *Ukash* para restaurar os arquivos e desbloquear a tela. É conhecido como um *malware* de vários componentes empacotado como um arquivo de autoextração (SFX). Pode ser empacotado com aplicativos de terceiros, como Sdelete e WinRAR.
ANDROIDOS LOCKER	Locker	Primeiro Ransomware móvel achado e utiliza rede *The Onion Router* (TOR), um serviço legítimo que permite conexões anônimas. Os usuários com dispositivos móveis afetados por este malware podem encontrar os arquivos armazenados em seus dispositivos móveis inutilizados e detidos para resgate.
CRIBIT (*BitCrypt*)	Crypto	Semelhante ao CRILOCK com o uso da criptografia RSA-AES para arquivos de destino, a versão 1 usa RSA-426, a versão 2 usa RSA-1024. Acrescenta a string bitcryp1 para a versão 1 e bitcrypt2 para a versão 2 no nome da extensão dos arquivos que criptografa.
CRILOK (*CryptoLocker*)	Crypto	Emprega Algoritmo de Geração de Domínio (DGA) por sua conexão de servidor C&C, em Outubro de 2013 foi encontrado como parte do correio de spam que faz o download do ZBOT, que ainda faz download do CRILOCK.
CRYPAURA (*PayCrypt*)	Crypto	Criptografa arquivos e anexa o contato de endereço de *e-mail* correspondente para decifragem de arquivo. A versão *PayCrypt* acrescenta .id aos arquivos que criptografa.
CRYPCTB (*CTB Locker*)	Crypto	Criptografa arquivos de dados e garante que não haverá recuperação de arquivos cifrados excluindo suas cópias sombra. Chega via spam que contém um anexo, na verdade um downloader deste *Crypto Ransomware*. Utiliza engenharia social para atrair usuários a abrir o anexo e utiliza Tor para mascarar suas comunicações C & C.[PCrisk , 2016]

Família	Tipo	Descrição
CRYPDEF (*CryptoDefense*)	Crypto	Para decrifrar arquivos, pede aos usuários que paguem dinheiro de resgate em moeda de *bitcoin*. O *CryptoDefense* também exclui todas as cópias do Shadow Volume quando é lançado, o que significa que a única maneira de restaurar os arquivos é através de um *backup*. Conecta-se a um servidor de Comando e Controle (C&C) e faz o *download* de uma chave privada.
CRYPTCOIN (*CoinVault*)	Crypto	Cifra arquivos e exige que os usuários paguem em bitcoin para descifrar arquivos. Oferece um teste gratuito único para decifrar um arquivo.
CRYPFILE	Crypto	Usa a chave pública única gerada RSA-2048 para criptografia de arquivos e também pede aos usuários que paguem 1 *bitcoin* para obter a chave privada para decifrar os arquivos.
CRYPWALL (*CryptoWall*)	Crypto	Relatado para ser a versão atualizada de *CryptoDefense*. Utiliza moeda bitcoin como modo de pagamento, utiliza a rede TOR para fins de anonimato, chega via spam. O CryptoWall 3.0 vem acompanhado de *spyware* FAREIT. O Cryptowall 4.0 criptografa o nome do arquivo que criptografa e segue uma nota de resgate atualizada, também chega por spam como um anexo de *JavaScript* e pode ser baixado por variantes de TROJ.KASIDET.
CRYPTOR	Crypto	Chega através de DOWNCRYPT, um arquivo em lote *Ransomware* capaz de cifrar arquivos de usuário usando o aplicativo GNU *Privacy Guard*.
VIRLOCK	Crypto	O Virlock é um desses casos especiais de *Ransomware* que criptografa arquivos e também os infecta. Como resultado, qualquer usuário que posteriormente abre o arquivo infectado também é infectado, fazendo com que todos os arquivos se tornem cifrados e infectados. A última variante do Ransomware Virlock normalmente chega através de compartilhamentos externos ou USB sticks. Em alguns casos, Virlock também foi observado empacotado com outro *malware*.

PGPCODER	Crypto	Descoberto em 2005, foi o primeiro Ransomware visto. Uma vez executado, o *malware* executa sua missão, que é cifrar, usando uma chave de criptografia digital, todos os arquivos encontrados em unidades de computador com extensões correspondentes às listadas em seu código. Essas extensões incluem .doc, .html, .jpg, .xls, .zip e .rar.
KOLLAH	Crypto	Um dos primeiros *Ransomwares* que criptografa arquivos usando certos nomes de extensão. Os arquivos de destino incluem documentos do Microsoft Office, arquivos PDF e outros arquivos considerados ricos em informações e relevantes para a maioria dos usuários. Adiciona a string GLAMOUR aos arquivos que criptografa.[Corporation , 2016]
MATSNU	Locker	*Backdoor* que tem capacidades de bloqueio de tela, pede resgate. Após a execução, ele modifica certos registros para permitir que suas cópias executem toda vez que o sistema inicializar e desabilita alguns dos processos, como o editor do registro e o gerenciador de tarefas. Ele também exclui determinados registros para desativar o usuário ao iniciar o computador no modo de segurança.
REVETON	Locker	Bloqueia a tela usando uma exibição falsa que adverte o usuário que eles violaram a lei federal. A mensagem declara ainda que o endereço IP do usuário foi identificado pelo FBI, como sites visitantes que apresentam conteúdo ilegal.
VBUZKY	Crypto	Ransomware de 64 bits, possui tentativas de usar a injeção Shell-TrayWnd e ativa a opção TESTSIGNING do Windows 7.
GULCRYPT	Crypto	Cria arquivos com extensões específicas e deixa um arquivo de texto de resgate contendo as instruções sobre com quem entrar em contato e como descompactar os arquivos que contêm arquivos do usuário.
CRYPDIRT	Crypto	Criptografa os bancos de dados no servidor da Web tornando o site indisponível e utiliza HTTPS para se comunicar com o servidor C & C. A chave de descodificação está somente disponível no servidor C & C.
CRYPWEB	Crypto	Visto pela primeira vez em 2013 antes do surgimento de *CryptoLocker*.

CRYPTORBIT	Crypto	Detecção de imagens, texto e arquivos HTML que contenham notas de resgate que são indicadores de comprometimento (COI). Destina-se a sistemas operacionais Windows e criptografa todos os arquivos no computador, independentemente da extensão deles.
CRYPTLOCK (*TorrentLocker*)	Crypto	Variantes mais recentes exibem "crypt0l0cker"no computador afetado. Utiliza uma lista de extensões de arquivo que evita cifrar, em comparação com o *Ransomware* usual que usa uma lista de extensões de arquivo para cifrar.
CRYPFORT (*CryptoFortress*)	Crypto	Imita a interface do usuário TorrentLocker / CRYPTLOCK e criptografa arquivos em pastas compartilhadas.
CRYPTESLA (*TeslaCrypt*)	Crypto	A interface do usuário é semelhante ao CryptoLocker. As versões 2.1 e 2.2 anexam arquivos cifrados com .vvv e .ccc. A versão 3.0 possui um algoritmo de criptografia aprimorado e acrescenta .xxx, .ttt e .mp3 aos arquivos que criptografa.[Scaife]
CRYPVAULT (*VaultCrypt*)	Crypto	IUsa a ferramenta de criptografia GnuPG e ferramenta de hacking de downloads para roubar credenciais armazenadas em navegadores da *Web*.
CRYPSHED (*Troldesh*)	Crypto	Visto pela primeira vez na Rússia. Adicionou tradução em inglês para sua nota de resgate para segmentar outros países. Além de anexar .xtbl ao nome do arquivo cifrado, ele também codifica o nome do arquivo, fazendo com que os usuários afetados percam o controle de quais arquivos são perdidos.
SYNOLOCK (*SynoLocker*)	Crypto	Explora o sistema operacional dos dispositivos NAS da Synology DSM 4.3-3810 ou anterior, para cifrar arquivos armazenados nesse dispositivo. Tem um portal de suporte ao cliente.
KRYPTOVOR	Crypto	Parte de uma infecção de vários componentes. Além de seu componente *Crypto Ransomware*, utiliza uma biblioteca de código aberto de Delphi chamada LockBox 3 para cifrar arquivos.

CRYPFINI	Crypto	Chega via *spam*, o correio de *spam* geralmente finge ser um aplicativo de trabalho vinculado a uma publicação. Acrescenta arquivos .crinf.
CRYPRADAM (*Radamant*)	Crypto	Acrescenta .rdm aos arquivos que criptografa. Este *malware* verifica todos os arquivos que combinam certas extensões e as criptografa usando uma chave única AES-256 para cada arquivo. A chave AES-256 gerada é então criptografada com uma chave mestre que é incorporada no arquivo de destino.
CRYPTRITU (*Ransom32*)	Crypto	Ransom32 é um novo *Crypto Ransomware* oferecido como um Serviço (RaaS). O *Ransomware* Ransom32 é escrito em *JavaScript*. Após infiltração, este *Ransomware* encripta os dados. Além disso, o proprietário do sistema infectado é solicitado a pagar um resgate em troca da chave de decifragem. Ransom32 RaaS permite aos criminosos virtuais distribuir e descarregar as suas cópias deste *Ransomware*. [PCrisk , 2016]
CRYPBOSS	Crypto	*CrypBoss* é uma família de *Ransomware* que visa o Windows. Os arquivos cifrados são renomeados para .crypt ou .R16M01D05. O *malware* derruba notas de resgate denominadas HELP_DECRYPT.jpg ou HELP_DECRYPT.txt em vários locais no sistema. As notas de resgate instruem-se a contactar um endereço de *e-mail*.
CRYPDAP (*PadCrypt*)	Crypto	Possui suporte ao bate-papo ao vivo para usuários afetados. Chega via spam. O PadCrypt encripta vários tipos de diretórios, tais como fotos, vídeos e por assim em diante, usando o algoritmo de encriptação AES-256. O PadCrypt é mais propenso a ser distribuído via *e-mail*, redes P2P (por exemplo, Torrent), atualizações de *software* falsos e trojans.

CRYPHYDRA (*HydraCrypt*)	Crypto	É um *Ransomware* que encripta a grande maioria dos diretórios armazenados no computador da vítima. HYDRACRYPT também acrescenta 'hydracrypt_ID_ [8 caracteres aleatórios]' extensão para cada ficheiro encriptado. Após a infiltração bem sucedida, *HydraCrypt* exibe uma mensagem informando que a vítima deve comprar o *software*, que foi desenvolvido pelos criminosos virtuais, que será usado para desencriptar os arquivos citados. [Digima , 2016]
LOCKY	Crypto	Renomeia arquivos cifrados para valores hexadecimais. Acrescenta .locky aos arquivos que criptografa. Chega via spam com anexos .DOC macro-embutidos, semelhante à chegada do *malware* DRIDEX.
REX	Locker	Rex é um Linux *Ransomware* que surgiu inicialmente em maio de 2016 e foi encontrado visando sites do Drupal com administradores alegando que seus sites estavam sendo bloqueados. A nova versão do Rex é capaz de transformar sistemas infectados em bots que ele usa para executar ataques *Distributed Denial of Service*(DDoS). O Rex lança plugins de Controle de Procedimento Remoto (RPC) e verifica vulnerabilidades no *software* comum do servidor Linux, como o *scanner* DrupalRESTWS,, WordPress, ContactScanner, Magento, Kerner, Airos, Exagrid, Jetspeed e o RansomScanner.[acunetix , 2016]
SATANA	Locker	O malware intitulado Satana ("Satã"), de origem russa. Ele executa duas tarefas: bloqueia arquivos e corrompe o *Master Boot Record* (MBR) do Windows, bloqueando o processo de *boot* do Sistema Operacional.[Kaspersky , 2016]
VIPASANA	Crypto	A mensagem de resgate deste malware é em russo e cria arquivos maliciosos no PC do usuário, após o qual verifica e criptografa os dados do usuário com um código forte. É distribuído através de URLs mal-intencionados ou anexos maliciosos.[Semvirus , 2016]

CRYPTXXX	Crypto	Os arquivos do computador e de qualquer mídia de armazenamento são cifrados pouco depois da infecção. O *malware* cria três arquivos: um de texto, uma imagem e uma página HTML. A imagem é colocada como papel de parede da área de trabalho. A página na *web* é aberta no navegador, enquanto o arquivo de texto é mantido no disco rígido. Todos os manuais têm textos similares.[Kaspersky , 2016]
ALPHACRYPT	Crypto	AlphaCrypt *Ransomware* é uma nova versão do TeslaCrypt que se alastra juntamente com anexos de *e-mails* de *spam* e, uma vez dentro, bloqueia arquivos importantes sem deixar qualquer hipótese de os recuperar, a não ser que pague pela chave de decifragem. AlphaCrypt *Ransomware* examina os discos do seu computador e procura por certas extensões de arquivos. Os arquivos necessários encontrados, são cifrados usando a extensão .ezz.
GRYPHON	Crypto	GRYPHON encripta dados armazenados e acrescenta nomes de diretórios com a extensão ".[Test].gryphon". Outras variantes usam a extensão ".[decr@cock.li].gryphon"para os diretórios cifrados. Após a encriptação bem-sucedida, o GRYPHON cria um ficheiro de texto "!!!DECRYPT FILES ##!.txt"contendo uma mensagem de resgate e coloca-a em cada pasta que contém dados cifrados.
CRYPMIC	Crypto	CrypMIC é um *malware* de tipo semelhante ao Ransomware CryptXXX e UltraCrypter. Na sequência de infiltração, CrypMIC encripta vários dados armazenados no sistema usando um algoritmo de encriptação assimétrica. Ao contrário de outras infecções de *malware* semelhantes, CrypMIC não acrescenta o nome de diretórios cifrados com uma extensão específica, o que torna mais difícil determinar os dados que foram comprometidos.
CRYPTOSHIELD	Crypto	CryptoShield é uma versão atualizada do Ransomware CryptoMix. CryptoShield é distribuído usando Exploit Kits. Seguindo a infiltração, este *Ransomware* encripta vários dados usando a encriptação RSA-2048 e acrescenta a extensão ".CRYPTOSHIELD"ao nome de cada ficheiro. Assim que a encriptação é bem sucedida, são criados dois arquivos (" RESTORING FILES .HTML"e " RESTORING FILES .TXT"), e os coloca em cada pasta que contém dados cifrados.[PCrisk , 2017]

CHIP	Crypto	O *CHIP Ransomware* é um *malware* que está sendo distribuído usando o RIG Exploit Kit. Este *malware* criptografa os arquivos da vítima para que eles não sejam mais acessíveis. O CHIP exigirá o pagamento de um resgate depois de deixar um arquivo de texto chamado 'CHIP_FILES.txt', que alerta a vítima do ataque e instrui a vítima sobre como pagar o resgate. As vítimas são convidadas a pagar através da rede TOR para permanecer anônimo, e são convidados a deixar um número de identificação e uma mensagem.[Software , 2017]
CERBER	Crypto	O *Cerber Ransomware* adiciona a extensão .CERBER em cada arquivo que o *Cerber Ransomware* criptografa. Depois de ter cifrado alguns dos arquivos da vítima, este *malware* exige o pagamento de um resgate em troca da chave de decifragem. De acordo com o pedido de resgate do Cerber Ransomware, os usuários de computador têm uma semana para pagar o valor do resgate, antes que este montante dobre..[PCrisk , 2017]
CRYPSPAM	Crypto	Usa explorações na aplicação de servidor de código aberto JexBoss e em outras plataformas de aplicativos baseadas em Java para instalar-se em servidores de aplicativos da *Web* direcionados.
PETYA	Crypto	O Petya utiliza uma carga útil que infecta a gravação de inicialização principal (MBR) do computador, substituindo o carregador de inicialização do Windows e, em seguida, desencadeando uma reinicialização. Na próxima inicialização, a carga útil é executada, e então criptografa a tabela de arquivos mestre do sistema de arquivos NTFS e, em seguida, exibe a mensagem de resgate exigindo um pagamento feito por *Bitcoin* .
WALTRIX	Crypto	Chega como um arquivo .DLL. Distribuído pelo Angler Exploit Kit, bloqueia telas e criptografa todos os arquivos e acrescenta a extensão .crypt.
CRYPSALAM	Crypto	Criptografa arquivos e descarta uma nota de resgate. Solicita aos usuários que contatem o criador do *Ransomware* por *e-mail* para decifrar os arquivos..

JIGSAW	Crypto	Exclui arquivos e aumenta a quantidade de resgate a cada hora. Algumas variantes têm suporte ao bate-papo ao vivo para as vítimas. Quando o *Ransomware* Jigsaw é iniciado, ele irá verificar suas unidades para determinadas extensões de arquivo, criptografá-las usando criptografia AES e anexar um .FUN, .KKK, .GWS, ou, .Extensão BTC ao nome do arquivo dependendo da versão.
APOCALYPSE	Crypto	Acrescenta a extensão .encrypted aos arquivos cifrados. Exige que a vítima envie um *e-mail* ao atacante para obter instruções de resgate.
STAMPADO	Crypto	Tem semelhanças com as notas de resgate JIGSAW. Acrescenta a extensão .locked aos arquivos que foram cifrados. Criação de arquivos na pasta do usuário atacado com nomes compostos por caracteres hexadecimais.

Em praticamente todos *Ransomwares* há alguma tática de engenharia social: a mensagem é acompanhada de um argumento que tenta te convencer a clicar em um link ou anexo que leva ao *Ransomware*. O texto pode dizer, por exemplo, que a vítima tem uma dívida não paga, uma pendência com a justiça, uma atualização de segurança do banco, persuadindo a pessoa a clicar no link ou anexo.

Por muitos anos, o envio de e-mail de spam usando temas de engenharia social foi o método de escolha para distribuir todos os tipos de *malware*, incluindo *Ransomware*. Os cibercriminosos usam uma botnet para enviar o spam. O *spam* geralmente vem na forma de um *e-mail* contendo um anexo malicioso ou um *link* em que leva o usuario a um site contendo um *explit kit* . Os *e-mails* de *spam* incorporaram toda uma de engenharia social para enganar os usuários na instalação do *Ransomware*.

Após a contaminação, o *Ransomware* parte a ação impeditiva, ou seja, executa as instruções que bloqueia ou criptografa o sistema inteiro ou um conjunto de arquivos. Essa ação é executada dependendo da forma de como o *malware* foi escrito e dos recursos explorados.

2.3 Vulnerabilidades Exploradas

Os *Ransomwares* não permitem acesso externo ao computador infectado como os trojans, a maioria é criada com o propósito comerciais, geralmente são detectados pelos antivírus com uma certa facilidade, pois costumam gerar arquivos cifrados grandes, embora alguns possuam opções que escolhem inteligentemente quais pastas cifrar, ou então, permitem que o atacante escolha quais as pastas de interesse. Os principais alvos desse tipo de malware são usuários domésticos, empresas de negócios e agências públicas.

Ransomware seja talvez o malware mais eficaz contra indivíduos que não são tão familiari-

zados com uso de computadores e internet, pois não percebem o risco de serem atacados por algum *malware*. O grupo mais comum que é afetado por *Ransomware* é o usuário doméstico, que freqüentemente tem menos acesso à assistência técnica e suporte de TI. A falta de suporte especializado faz com que o usuário se sinta mais ameaçado e tenha uma atitude de pagar para ter o seu dispositivo/dados desbloqueado.

Em geral, usuários domésticos têm informações, arquivos e documentos pessoais valiosos no computador, como: projetos de faculdade, fotos e arquivos de gravação de vídeo. Apesar de serem dados de valor para utilizadores domésticos, é pouco provável que eles disponham de uma estratégia de *backup* eficaz para eventos como um incêndio ou roubo, muito menos um ataque de *Ransomware*. Uma pesquisa da Symantec/Norton mostrou que 25% dos usuários domésticos não faziam nenhum *backup* em qualquer tipo de arquivo de seu computador pessoal. 55% realizava *backup* de alguns arquivos. Apenas 25% dos usuários realizava *backup* de arquivos uma vez por semana. O restante fazia apenas backups uma vez por mês ou mesmo menos frequentes. Mesmo que certo usuário doméstico tenha um processo de *backup* eficiente, alguns ataques excluem cópias salvas locais do computador e criptografam arquivos em dispositivos de armazenamento externos conectados ao computador [Savage; Coogan; Lau, 2015].

Para muitas empresas, a informação e a tecnologia são indispensáveis para a realização de tarefas diariamente. Os computadores de negócios também são mais suscetíveis a conter dados importantes, como: bancos de dados de clientes, planos de negócios, propostas, relatórios, códigos fonte, formulários. As ameaças modernas do *Crypto Ransomware* podem acessar e enumerar todas as movimentações acessíveis, tais como o arquivos locais ou arquivos de servidores compartilhados [Alecrim, 2016].

As informações que são perdidas podem ter um impacto catastrófico no negócio da empresa. Embora muitas empresas tenham backup e planos de recuperação de acidentes, ainda há muitas que não. Alguns planos de recuperação de desastres da organização podem não abranger os utilizadores finais individuais. Esses fatores tornam as empresas um alvo viável para o *Ransomware* do tipo *Crypto*. [UBM , 2016]

Além do *Ransomware* que afeta os utilizadores empresariais individuais, também houve casos em que a própria empresa tinha sido alvo de *Ransomware* de criptografia de arquivos.

As principais vulnerabilidades exploradas pelos atacantes são:

- *Phishing* por *e-mails*; [Arcon , 2017]
- Engenharia social por canais *web*;
- URLs maliciosas;
- Bugs/Atualizações em *softwares* reconhecidos globalmente (Ex.: Adobe, Microsoft Sil-verlight)
- Aplicativos maliciosos em plataforma Android;
- Exploração do certificado iOS (Apple) para transmitir conteúdo malicioso.

Após ter sido autenticado em um dispositivo, o atacante terá todo acesso a discos físicos que estejam conectados naquele momento e além disso, as unidades de rede compartilha-das são igualmente vulneráveis. Os atacantes sabem que empresas e grandes corporações es-tão utilizando o armazenamento de dados em nuvem e criaram *Ransomware* que atinge arquivos mantidos na nuvem.[Cybereason , 2017]

O *Ransomware* tem como alvo diversos tipos de arquivos, sendo alguns mais óbvios como: fotos (jpg, png), arquivos salvos do pacote Office (doc, ppt, xls). E algumas extensões menos óbvias, como: css, java, javascript, tiff, php, wav. Pode infectar computadores (desktop, notebook, servidores), equipamentos de rede (modens, switches, roteadores) e dispositivos móveis (tablets, celulares, smartphones) [Savage; Coogan; Lau, 2015]. Atualmente, os *Ransomwares* que estão associados aos ataques, atingem qualquer tipo de arquivo, independente de sua extensão.

2.3.1 Ataques a Computadores Pessoais

A grande maioria das ameaças de *Ransomware* hoje são projetados para alcançar computadores pessoais executando o sistema operacional Windows. Isso não é surpreendente, pois os computadores baseados no Windows representam cerca de 89% dos SO no mercado para computadores, com o Mac OS X e Linux representando o restante [Savage; Coogan; Lau, 2015].

Ransomwares tem de ser adaptados especificamente para um determinado sistema operacional, porque muitas vezes tem de alavancar uma API para bloquear ou limitar o acesso a controles, como o mouse/teclado. Além disso, muitas ameaças de Ransomware agora fazem uso de bibliotecas de criptografia embutidas ou APIs fornecidas com o sistema operacional para executar o processo de criptografia e decifragem propriamente dito. Isso evita que os invasores inventem sua metodologia de criptografia e propagação de arquivos e bibliotecas adicionais com o seu malware.

No entanto, em reconhecimento do pequeno, mas significativo conjunto de utilizadores não-Windows, alguns cibercriminosos criaram o Trojan Browlock. A ameaça é implementada em JavaScript e é projetada para trabalhar em uma ampla gama de navegadores *web*, independentemente do Sistema Operacional. Embora esta técnica de bloqueio de navegador seja menos eficaz do ponto de vista técnico, esta tática é projetada para atingir as vítimas que não podem ser alvejados de outra maneira.

2.3.2 Ataques a Dispositivos Móveis

Os dispositivos móveis mais visados pelo *Ransomware* são *tablets* e telefones celulares. Estes dispositivos tornaram-se indispensáveis mundialmente, com estudos mostrando que os usuários estão gastando mais tempo em dispositivos móveis do que nunca. Hoje, existem basicamente apenas dois principais concorrentes no mercado de SO móvel: Android e iOS.

Os usuários do iOS que não realizaram "jail-break" são bem protegidos pelo sistema bem controlado da Apple. Para um usuário que não realizou o "jail-break", sua capacidade de instalar aplicativos fora da App Store oficial é extremamente limitado. Isto faz com que a atividade de

desenvolvimento de *Ransomwares* para iOS seja muito arriscado, com pouca perspectiva de retorno.

O Android é uma plataforma muito mais aberta e permissiva. Apresenta vantagens e desvantagens. Muitos usuários gostam da liberdade e flexibilidade para escolher instalar qualquer tipo de app que desejam de qualquer fonte. A desvantagem é que esta mesma flexibilidade pode torná-la mais fácil para os criadores de *malware* para operar e espalhar seus *malwares*. Isto é uma das principais razões que ameaças em Android são muito mais comuns quando comparada com IOS.

O Android.Fakedefender, descoberto em junho de 2013, marcou o cruzamento do padrão de um falso antivírus para o *Ransomware* tipo *Locker* na plataforma Android.Fakedefender pretendia ser um *scanner* de segurança, mas quando inevitavelmente encontrasse "ameaças críticas", a interface do dispositivo era bloqueada para evitar que as vítimas executassem outros aplicativos ou alterassem configurações no sistema operacional. Estas táticas foram todas concebidas para coagir as vítimas para pagar uma licença para o *software* falso, uma vez que o *Ransomware* garante em resolver os problemas relatados. [Netsafe , 2015]

Posteriormente os atacantes começaram a concentrar-se puramente em ser de fato um *Ransomware* tipo Locker do que fingir ser uma ferramenta de segurança. Android.Lockdroid.E, visto em 2014, foi um dos primeiros exemplos desta classe de atacar dispositivos Android.

2.3.3 Ataques a Servidores

Servidores representam um diferente tipo de proposta para os atacantes que visam extorquir dinheiro de suas vítimas. Servidores são muito mais propensos a conter dados sigilosos e importantíssimos da parte operacional de uma empresa. Eles agem como repositórios centrais para documentos, códigos-fonte, recibos financeiros, transações, base de dados de usuários, tornando-os assim alvos de grande valor. Dado o papel crítico que os servidores desempenham, muitas organizações empresariais possuem planos de recuperação por desastres e "continuidade do negócio"que circundam as operações e garantem o backup dos dados. Apesar disso, fazer com que um servidor fique inativo, mesmo que por curto período de tempo, pode ser absolutamente prejudicial. Em razão desses planos de contingência que as empresas possuem, os atacantes vem sendo forçados a adotar diferentes técnicas de extorquir as organizações empresariais quando atacam seus servidores.

As empresas de segurança da informação tem observado que os atacantes tradicionalmente chantageiam as empresas ao desencadear um ataque de negação de serviço (DDoS) nos servidores das empresas e em seguida iniciam o processo de extorsão. Como resultado disso, muitas empresas que são suscetíveis à ataques DDoS tem solicitado a ajuda de serviços de mitigação de DDoS para reduzir o impacto destes ataques. Isso tem encorajado os atacantes a procurar formas alternativas de forçar as empresas a pagarem o resgate, atingindo o servidor - um de seus ativos mais importantes de infraestrutura de redes - e os dados que ele contém.

Alguns grupos de atacantes fazem isso infiltrando no servidor alvo e corrompendo o *software*. Dessa forma os dados armazenados ficam cifrados de uma forma que somente os atacantes possuem a chave para decifrar. A premissa do ataque é cifrar silenciosamente todos os dados contidos no servidor, incluindo os *backups*. Esse é um processo que leva tempo e, dependendo da organização,

requer paciência dos atacantes para que o ataque dê certo. Quando os arquivos e *backups* suficientes já estão cifrados, os atacantes removem a chave de decifragem e fazem o pedido do resgate dos arquivos envolvendo dinheiro ou *bitcoins*.

No trabalho foi utilizado uma máquina virtual da Oracle, em que foi instalado o sistema operacional Windows XP SP3, com a finalidade de servir de alvo para realizar ataques e instrumento de análise das diversas famílias de *Ransomwares*. Foi escolhido o Windows XP SP3 devido a sua praticidade e vulnerabilidade contra ameaças externas, o que indica que uma vez infectado pelo Ransomware, a probabilidade êxito para análise seria maior.

De acordo com um relatório da Microsoft de 2013 [Demartini , 2013], o Windows XP é o mais infectado por malwares. Esse relatório revelou que 9,1% dos computadores que utilizam Windows XP estão infectados com algum tipo de *malware*. A constatação aparece no *Security Inteligent Report*, documento que avalia o estado da segurança em tecnologia e toma como base dados enviados pelos antivírus ou ferramentas de remoção de *malware*. Em segundo lugar ficou o Windows Vista, com 5,5% de infecções, seguido do Windows 7, com 4,9%, e 8, com 1,6% de máquinas infectadas. Os dados são correspondentes ao período entre janeiro e junho de 2013 e compreendem mais de um bilhão de computadores, cujos usuários concordaram em compartilhar informações de segurança com a Microsoft.

Capítulo 3

Arquitetura Desenvolvida e Ferramentas Utilizadas

Para realizar a análise dos *Ransomwares* selecionados, as amostras deste *malware* foram executadas em uma máquina virtual via *VirtualBox*, denominada de *Guest*, com o sistema operacional Windows XP. A execução para análise da amostra é sempre controlada pela máquina física, denominada *Host*, com o sistema operacional Linux Ubuntu. O *Host* controla a execução com a ferramenta *Cuckoo Sandbox*, e as informações entre as duas máquinas são trocadas através de uma rede virtual, utilizando adaptadores de rede virtuais. O acesso à Internet pela máquina *Guest* é feito utilizando um adaptador de rede virtual da máquina *Host*, chamado Vboxnet0, que age como *Gateway*, redirecionando os pacotes destinados à máquina *Guest*. A Figura 3.1 ilustra o ambiente montado para execução das amostras e extração de características.

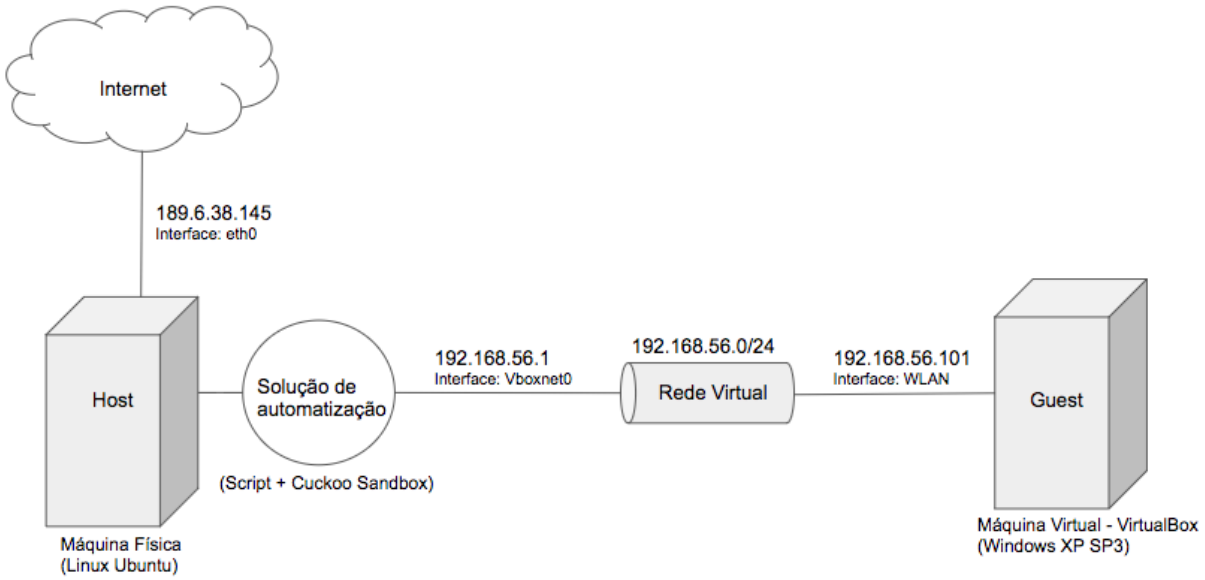

Figura 3.1: Arquitetura do ambiente construído neste trabalho para análise das amostras

Para que a análise fosse completamente automatizada, foi criado um script em *Shell Script* em que é necessário apenas indicar o caminho deste e o caminho do *malware* a ser analisado. O *script* tem a função de ligar a máquina virtual e deixá-la em uma instância limpa para que a análise do *Ransomware* seja realizada. Além disso, o código criado realiza o envio do arquivo malicioso para o *Cuckoo Sandbox* e que ao finalizar a execução do *Ransomware*, o relatório no formato HTML é gerado e apresentado ao usuário utilizando qualquer navegador *Web* que esteja instalado. O script utilizado para estas funções está comentado e disponível nos anexos deste documento. Na figura 3.2 é apresentado um exemplo de como um *malware* foi executado através do script criado.

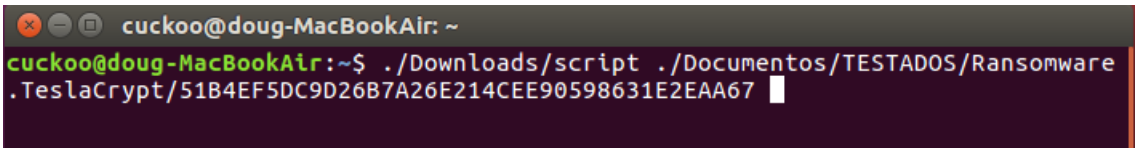

Figura 3.2: Terminal mostrando como realizar a execução do *malware* através do script

As análises realizadas foram configuradas para serem finalizadas após 120 segundos do início da execução do *malware* na VM.

3.1 *Cuckoo Sandbox*

Cuckoo Sandbox é um *software* gratuito que automatiza a tarefa de analisar qualquer arquivo mal-intencionado no Windows, OS X, Linux e Android, ou seja, *Cuckoo Sandbox* é um sistema de análise de *malware*. Pode-se lançar qualquer arquivo suspeito nele e, em questão de segundos, *Cuckoo* irá fornecer-lhe alguns resultados detalhados, descrevendo o que esse arquivo fez quando executado dentro de um ambiente isolado. [Cuckoo Foundation, 2016].

Devido ao extenso design modular do *Cuckoo*, é possível personalizar as etapas de processamento e relatórios. O *Cuckoo* fornece todos os requisitos para integrar o *Sandbox* em suas estruturas e armazenamentos existentes com os dados desejados, da forma desejada, com o formato desejado. Esse *software* é capaz de:

- Analisar muitos arquivos maliciosos diferentes (executáveis, *applets* Java), bem como sites mal-intencionados, em ambientes virtualizados Windows, OS X, Linux e Android;
- Descarregar e analisar o tráfego de rede, mesmo quando criptografado;
- Capturas de tela no momento da análise de uma amostra;
- Executar uma análise de memória avançada do sistema virtualizado infectado com suporte integrado para a ferramenta *Volatility*;
- Traçar chamadas e comportamento geral de arquivos.

A ferramenta do *Cuckoo* consiste em gerenciar a execução de uma amostra selecionada e realizar as análises nas máquinas virtuais que executam as amostras. Para cada execução que é iniciada,

uma instância da máquina virtual limpa e isolada é criada, pois é gerado um *snapshot* antes da execução de qualquer *malware*. A Figura 3.3 ilustra uma rede virtual criada para ser utilizada com a ferramenta *Cuckoo Sandbox*.

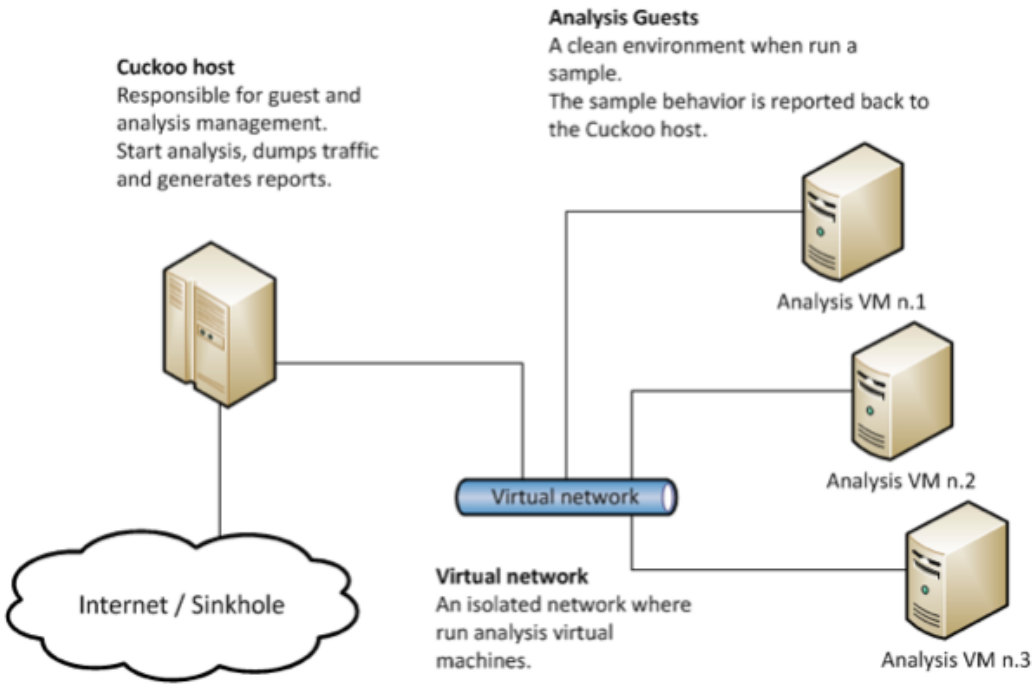

Figura 3.3: Arquitetura do *Cuckoo Sandbox* [Cuckoo , 2017]

Neste trabalho, foi utilizado uma máquina virtual denominada como o alvo do envio da amostra de *Ransomware*, mas também é possível realizar essa análise utilizando máquinas físicas, em que são utilizadas imagens dessas máquinas para que se tenha instâncias limpas no sistema operacional.

Além disso, existem módulos que são de extrema importância para o funcionamento correto do *Cuckoo Sandbox*. Estão estruturados na linguagem *Python* e indicam como as informações devem ser coletadas, processadas e apresentadas. Existem módulos padrão fornecidos em conjunto com a ferramenta, mas também existe a possibilidade de realizar a criação de módulos específicos para atender as demandas do usuário. Os três principais tipos de módulos são processamento, assinatura e relatório.

3.1.1 Módulos de Processamento

Os módulos de processamento do *Cuckoo* são *scripts* que permitem definir maneiras personalizadas de analisar os resultados brutos gerados pelo *Sandbox* e anexar algumas informações a um contêiner global que será usado posteriormente pelo módulo de assinaturas e pelo módulo de relatórios, que serão explicitados adiante. Esses módulos estão localizados na pasta /.cuckoo/modules/processing. A figura 3.4 demonstra um exemplo básico de como é um módulo de processamento.

```
1  from cuckoo.common.abstracts import Processing
2
3  class MyModule(Processing):
4
5      def run(self):
6          self.key = "key"
7          data = do_something()
8          return data
```

Figura 3.4: Exemplo de código do módulo de processamento do *Cuckoo Sandbox*. [Cuckoo , 2017]

Os módulos de processamento disponíveis por padrão do *Cuckoo Sandbox* são:

Tabela 3.1: Módulos de Processamento do *Cuckoo Sandbox*

Módulo	Descrição
AnalysisInfo (cuckoo/processing/analysisinfo.py)	Gera algumas informações básicas sobre a análise atual, versão do *Cuckoo* e etc.
ApkInfo (cuckoo/processing/apkinfo.py)	Gera algumas informações básicas sobre a APK atual (análise de Android).
Baseline (cuckoo/processing/baseline.py)	Resultados da baseline da informação recolhida .
BehaviorAnalysis (cuckoo/processing/behavior.py)	Analisa os registros de comportamento e executa algumas transformações e interpretações iniciais, incluindo o rastreamento completo dos processos, um resumo comportamental e uma árvore de processo
Buffer (cuckoo/processing/buffer.py)	Análise de buffer
Debug (cuckoo/processing/debug.py)	Inclui erros e analisys.log gerado pelo analisador.
Droidmon (cuckoo/processing/droidmon.py)	Extrai a API dinâmica chamada pelos logo de informação Droidmon.
Dropped (cuckoo/processing/dropped.py)	Inclui informações sobre os arquivos deixados pelo malware e despejados pelo Cuckoo.
DumpTls (cuckoo/processing/dumptls.py)	Referências cruzadas das chaves TLS extraídas do monitor e informações chave extraídas do PCAP para despejar um arquivo de segredo mestre.

GooglePlay (cuckoo/processing/googleplay.py)	Informações sobre a sessão de análise utilizando GooglePlay.
Irma (cuckoo/processing/irma.py)	Conector IRMA.
Memory (cuckoo/processing/memory.py)	executa Volatility em um despejo de memória total.
Misp (cuckoo/processing/misp.py)	conector MISP..
NetworkAnalysis (cuckoo/processing/network.py)	Analisa o arquivo PCAP e extrai algumas informações de rede, como tráfego DNS, domínios, IPs, solicitações HTTP, tráfego IRC e SMTP.
ProcMemory (cuckoo/processing/procmemory.py)	Executa a análise do despejo da memória do processo.
ProcMon (cuckoo/processing/procmon.py)	Extrai eventos da saída de procmon.exe.
Screenshots (cuckoo/processing/screenshots.py)	Captura de tela e análise de OCR.
Snort (cuckoo/processing/snort.py)	Módulo de processamento Snort.
StaticAnalysis (cuckoo/processing/static.py)	Executa algumas análises estáticas dos arquivos PE32.
ProcMon (cuckoo/processing/procmon.py)	Extrai eventos da saída de procmon.exe.
Strings (cuckoo/processing/strings.py)	Extrai strings do arquivo binário analisado.
Suricata (cuckoo/processing/suricata.py)	Módulo de processamento Suricata.
VirusTotal (cuckoo/processing/virustotal.py)	Realiza pesquisa no VirusTotal.com para assinaturas de antivírus do arquivo analisado.

Os módulos de processamento possuem acesso a um conjunto de variáveis que contem os caminhos dos arquivos gerados pela análise, com diversos dados capturados da máquina virtual. Essas variáveis podem ser observadas na Tabela 3.2.

Tabela 3.2: Variáveis de ambiente disponíveis para módulos de processamento

Variável	Descrição
self.analysis_path	Caminho para a pasta que contém os arquivos de resultado da análise (storage/analyses/3/, por exemplo). Em que 3, é o número da análise realizada.
self.dropped_path	Caminho para a pasta que contém os arquivos baixados.
self.file_path	Caminho para o arquivo analisado.
self.log_path	Caminho para o arquivo analysis.log.
self.logs_path	Caminho para a pasta que contém os logs comportamentais brutos.
self.memory_path	Caminho para o arquivo de dump de memória.
self.pcap_path	Caminho para o dump de rede, no formato pcap.
self.pmemory_path	Caminho para o arquivo de dump de memória de processos.
self.shots_path	Caminho para a pasta que contém as capturas de tela.

3.1.2 Contêiner Global

Todo módulo que é inicializado e executado, os dados retornados serão anexados em uma estrutura de dados que chamaremos de contêiner global, que é simplesmente um grande dicionário de *Python* que inclui os resultados extraídos produzidos por todos os módulos classificados por sua chave de identificação. A ferramenta *Cuckoo* já fornece um conjunto de módulos padrão que gerará um contêiner global padrão.


```
{
    "info": {
        "added": 1506004454.651634,
        "started": 1506004455.088967,
        "duration": 142,
        "ended": 1506004597.122573,
        "owner": null,
        "score": 6.0,
        "id": 34,
        "category": "file",
        "git": {
            "head": "",
            "fetch_head": ""
        },
        "monitor": "2bd01ede5c5258d5fce2e38bc58348a62c11ce33",
        "package": "exe",
        "route": "none",
        "custom": null,
        "machine": {
            "status": "stopped",
            "name": "cuckoo1",
            "label": "WIN XP SP3 PRO",
            "manager": "VirtualBox",
            "started_on": "2017-09-21 14:34:15",
            "shutdown_on": "2017-09-21 14:36:36"
        },
        "platform": "windows",
        "version": "2.0.3",
        "options": "enable-services=False,procmemdump=yes,route=none"
    },
```


Figura 3.5: Estrutura de dados resultante do módulo analysisinfo.py.

3.1.3 Módulo de Asssinaturas

Os módulos de assinatura são utilizados para tratar os resultados gerados pelos módulos de processamento. Eles permitem a criação de assinaturas que podem identificar padrões predefinidos, que representam comportamentos específicos de amostras de *malware*. Esses módulos estão localizados na pasta /.cuckoo/modules/signatures.

Exemplos do que os módulos de assinatura podem detectar incluem:

- Identificar modificações feitas pela amostra de *malware* no sistema, como instalação de *drivers*.
- Identificar chamadas de APIs, buscando chamadas de APIs específicas.
- Acessar os arquivos criados e acessados por uma amostra de *malware*.
- Acessar os processos criados ou relacionados a uma amostra de *malware*.

Assim como nos módulos de processamento, a ferramenta *Cuckoo Sandbox* desenvolveu diversas assinaturas que foram utilizadas neste trabalho para analisar as características de comportamento dos *Ransomwares*. Na figura 3.6 é ilustrado um exemplo básico de assinatura padrão.

```
1  from cuckoo.common.abstracts import Signature
2
3  class CreatesExe(Signature):
4      name = "creates_exe"
5      description = "Creates a Windows executable on the filesystem"
6      severity = 2
7      categories = ["generic"]
8      authors = ["Cuckoo Developers"]
9      minimum = "2.0"
10
11     def on_complete(self):
12         return self.check_file(pattern=".*\\.exe$", regex=True)
```

Figura 3.6: Exemplo de código do módulo de assinaturas do *Cuckoo Sandbox* [Cuckoo , 2017]

3.1.4 Módulo de Relatórios

Os módulos de relatório são os módulos responsáveis por formatar e apresentar os resultados contidos no contêiner global. Após o processamento dos dados extraídos da máquina virtual pelos módulos de processamento e assinatura, o contêiner global de dados é passado para todos módulos de relatório disponíveis, que tem a tarefa de transformá-lo em informação acessível e de melhor visualização em diferentes formatos. No caso deste projeto, os arquivos de relatório gerados estão no formato JSON e HTML.[JSON , 2017] A figura 3.7 ilustra um exemplo de relatório HTML gerado neste projeto.

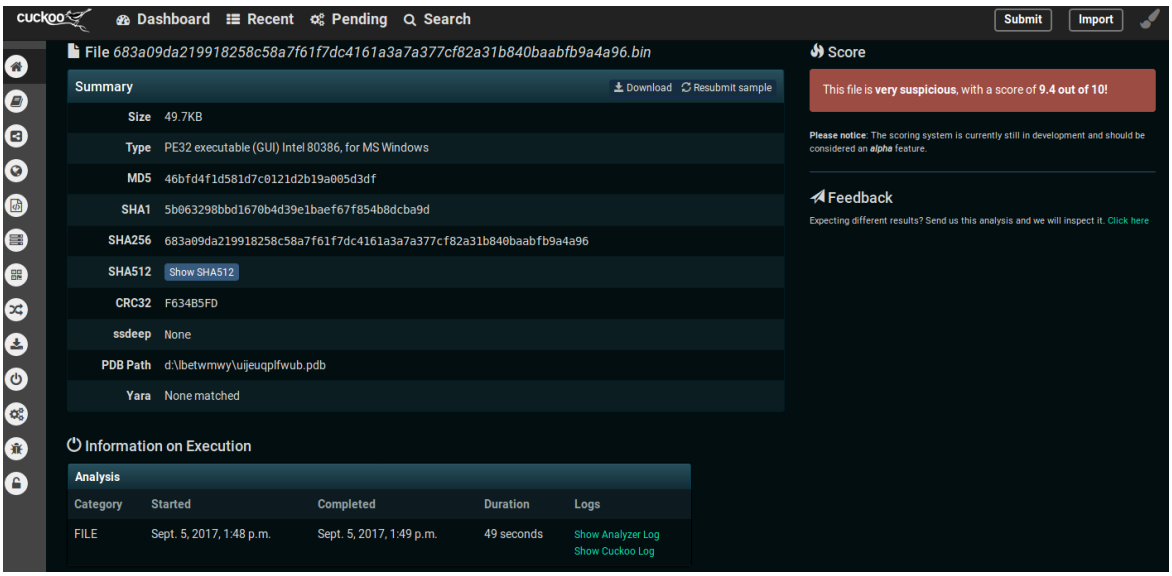

Figura 3.7: Exemplo de relatório HTML gerado após análise de um *Ransomware*

Os módulos de relatório são localizados na pasta /.cuckoo/modules/reporting, e assim como os módulos de processamento, devem ser habilitados para execução, no arquivo de configuração /conf/reporting.conf.

Ao terminar a execução da amostra, a ferramenta executa automaticamente todos os módulos que foram habilitados pelo usuário presentes na pasta /modules/reporting.

Esses módulos tem acesso à variáveis que contem informações relevantes acerca da análise realizada. As principais variáveis podem ser vistas na Tabela 3.3.

Tabela 3.3: Variáveis de ambiente disponíveis para o módulo de relatório

Variável	Descrição
self.analysis_path	Caminho para a pasta que contém os arquivos de resultado da análise (storage/analyses/3/, por exemplo). Em que 3, é o número da análise realizada.
self.reports_path	Caminho para a pasta onde os relatórios devem ser gravados (storage/analyses/3/reports/, por exemplo).
self.conf_path	Caminho para a pasta que contém o arquivo analysis.conf da análise em questão (storage/analyses/3/analysis.conf, por exemplo)
self.options	Um dicionário de dados contendo as opções especificadas pelo usuário em conf/reporting.conf.

3.1.5 Arquivos de configuração do *Cuckoo Sandbox*

Após realizar o download do *Cuckoo Sandbox*, algumas alterações foram necessárias para que a plataforma funcionasse corretamente . As alterações foram feitas nos arquivos cuckoo.conf, auxili-

ary.conf, processing.conf, virtualbox.conf e reporting.conf que se encontram no local /.cuckoo/conf.

A primeira configuração alterada foi no documento cuckoo.conf e que pode ser observada no código:

```
[resultserver]
# The Result Server is used to receive in real time the behavioral logs
# produced by the analyzer.
# Specify the IP address of the Host. The analysis machines should be able
# to contact the Host through such address, so make sure it's valid.
# NOTE: if you set resulserver IP to 0.0.0.0 you have to set the option
# resultserver_ip for all your virtual machines in machinery configuration.

ip = 192.168.56.1

# Specify a port number to bind the result server on.
port = 2042
```

O IP e porta inseridos foram necessários para que a arquitetura de comunicação do *Host* com a Rede Virtual que foi montada com a interface Vboxnet0 com IP 192.168.56.1 fosse mantida.

No arquivo auxiliary.conf foi necessário ativar o *sniffer* de rede, ou seja, a análise do tráfego de rede quando o *malware* for executado na máquina virtual. Após ser executado, um arquivo dump.pcap é gerado com os pacotes que trafegaram na rede durante a execução do *malware*. Caso não tenha tráfego de rede, o arquivo dump.pcap não será gerado. O parametro neste arquivo foi alterado de "no"para "yes".

```
[sniffer]

# Enable or disable the use of an external sniffer (tcpdump) [yes/no].
enabled = yes
```

No arquivo processing.conf, foi necessário ativar a ferramenta do VirusTotal, que é um serviço online que analisa arquivos e URLs, possibilitando a identificação de *malware* detectável por antivírus e scanners de websites.

```
[virustotal]

enabled = yes
```

No arquivo reporting.conf, foi ativado a ferramenta de geração de relatórios HTML após a execução de um malware na VM:

```
[singlefile]

# Enable creation of report.html and/or report.pdf?
enabled = yes
# Enable creation of report.html?
html = yes
```

As alterações seguintes foram feitas no arquivo virtualbox.conf. Primeiramente, foi inserido o nome da máquina virtual criada para análise de malware:

```
# Specify a comma-separated list of available machines to be used. For each
# specified ID you have to define a dedicated section containing the details
# on the respective machine.

machines = cuckoo1
```

Em seguida, foi necessário inserir o label correspondente a máquina virtual que foi criada:

```
[cuckoo1]

# Specify the label name of the current machine as specified in your
# VirtualBox configuration

label = WIN XP SP3 PRO
```

Também foi necessário inserir o sistema operacional ativo que está instalado na VM. Pode ser observada essa alteração que o sistema operacional da VM é Windows:

```
# Specify the operating system platform used by current machine
# [windows/darwin/linux]

platform = windows
```

A última alteração deste arquivo foi a inserção do IP da VM criada, que foi escolhido o IP 192.168.56.101:

```
# Specify the IP address of the current virtual machine. Make sure that the
# IP address is valid and that the Host machine is able to reach it. If not,
# the analysis will fail.

ip = 192.168.56.101
```

3.2 Máquina Virtual *VirtualBox*

Uma máquina virtual é um *software* de ambiente computacional em que um sistema operacional ou programa pode ser instalado e executado. De maneira mais simplificada, podemos dizer que a máquina virtual funciona como um "computador dentro do computador". [ZEBRA , 2012]

As máquinas virtuais podem proporcionar inúmeras vantagens sobre a instalação de sistemas operacionais e *softwares* diretamente no *hardware*. O isolamento, por exemplo, assegura que as aplicações e serviços que serão executados dentro de uma máquina virtual não poderão interferir no sistema operacional original e nem em outras máquinas virtuais.

Neste projeto foi utilizado uma VM (*Virtual Machine*) da *Oracle*, chamada VirtualBox. As máquinas virtuais foram utilizadas para criar um ambiente seguro e próprio para as análises do *malware* quando está sendo executado o *Cuckoo Sandbox*. [VirtualBox , 2017]

Na VM, está presente o sistema operacional Windows XP de 32 bits, em que foram instalados os programas: Google Chrome, Mozilla Firefox, Microsoft Office 2007 e Adobe Reader. Quando uma amostra de *Ransomware* é executada pelo *Cuckoo Sandbox* e é enviada à máquina virtual alvo, este *malware* possui interações com esses programas que foram instalados, como por exemplo, comandos para abertura do navegador para realizar requisições DNS para destinos aleatórios na Internet. Na figura 3.8, está sendo demonstrado a VM que foi criada para este projeto e suas especificações.

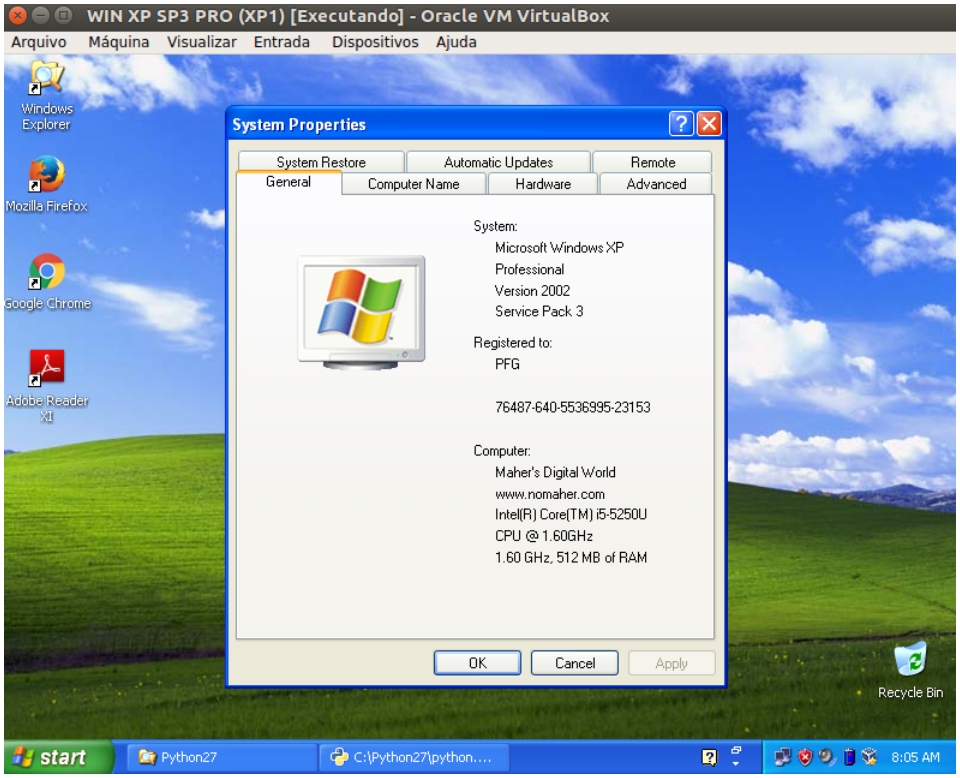

Figura 3.8: Informações da VM criada para utilização no projeto

Para que haja a integração do *Cuckoo Sandbox* com a máquina virtual em que deseja realizar a analise de uma amostra, é necessário que faça a instalação do Python no sistema operacional da máquina virtual. Neste trabalho, foi utilizado a versão do Python 2.7.4. [Python , 2017]. Na máquina *Host*, também foi instalado o Python e o arquivo agent.py foi copiado para o Windows XP.

O *Cuckoo* adota um agente personalizado agent.py que deve ser executado no Guest e que lida com a comunicação e a troca de dados com o Host. Para que *Cuckoo* funcione corretamente, é necessário instalar e iniciar esse agente. O Agente iniciará um pequeno servidor de API com o qual o *Host* poderá se comunicar.

Na VM utilizada, o endereço IP e DNS foram inseridos manualmente para que a arquitetura do ambiente construído para análise mostrada anteriormente fosse seguida. A figura 3.9 apresenta as configurações introduzidas na VM.

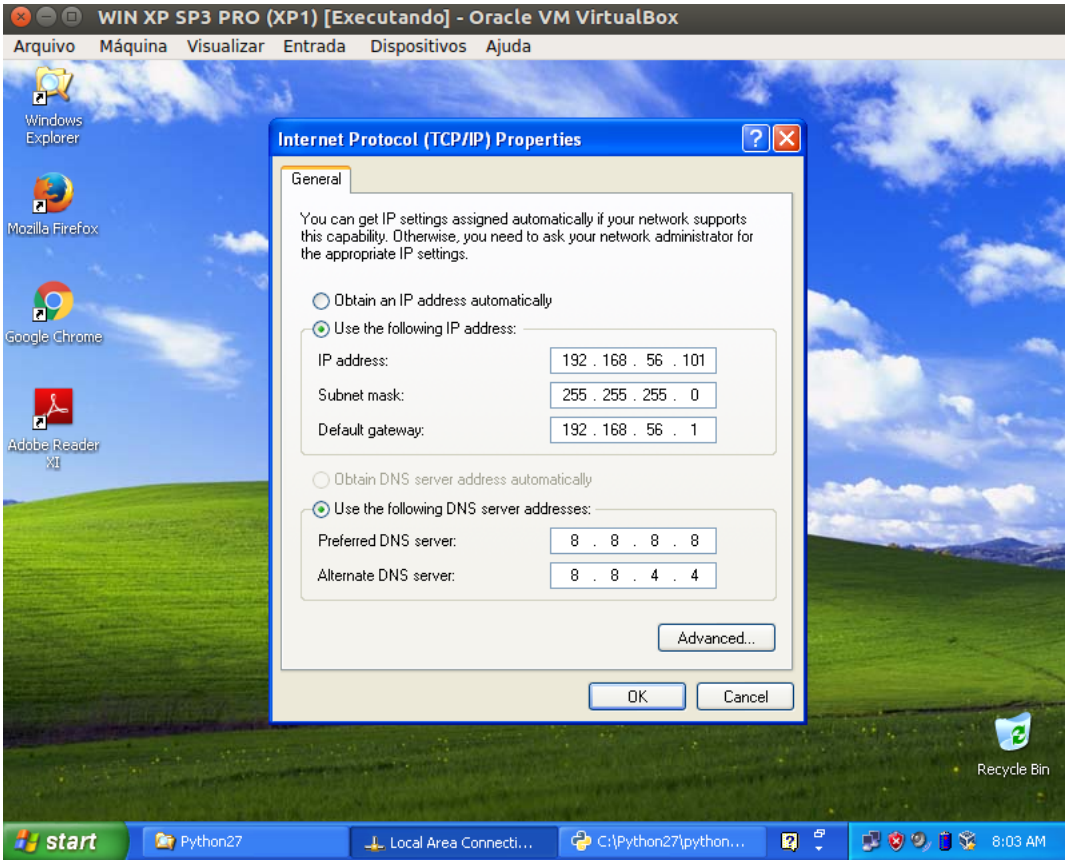

Figura 3.9: Configurações de IP e DNS fixos inseridos manualmente

Foi utilizado uma ferramenta do *VirtualBox* de *snapshot*, em que após a criação deste, é possível realizar a integração do *Cuckoo* com a VM dita anteriormente, que cria uma instância limpa da máquina para cada análise que deverá ser realizada. Caso o *snapshot* não seja criado, a plataforma do *Cuckoo Sandbox* não conseguirá detectar a instância para inicializar a análise do *malware*. O *snapshot* criado neste projeto pode ser observado na figura 3.1.

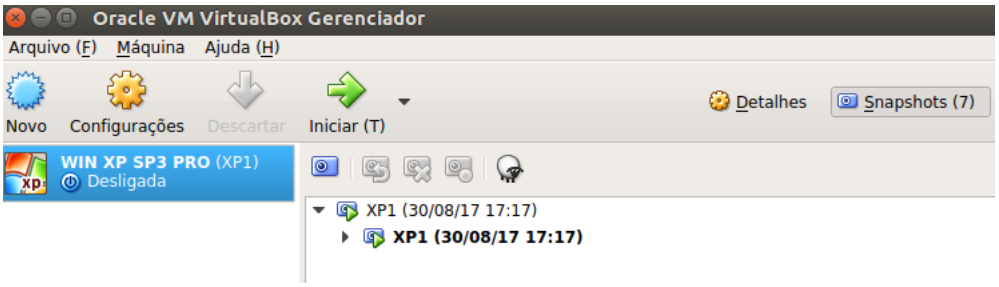

Figura 3.10: *Snapshot* XP1 criado na VM

3.3 *Wireshark*

O *Wireshark* é um analisador de pacotes de código aberto e gratuito. Ele é utilizado para solução de problemas de rede, análise, desenvolvimento de *software* e protocolo de comunicação. Ele é um programa de captura de dados que compreende a estrutura de diferentes protocolos de rede. Pode analisar e exibir os campos, juntamente com seus significados, conforme especificado por diferentes protocolos de rede.[Wireshark , 2017]

Esta ferramenta utiliza arquivos .pcap para capturar pacotes, portanto, ele só pode capturar pacotes nos tipos de redes que o pcap é suportado.

Neste projeto, o *Wireshark* foi utilizado para analisar o tráfego da rede da máquina virtual enquanto a amostra enviada pelo *Cuckoo* estava sendo executada. Quando a execução é finalizada, pode-se encontrar o arquivo dump.pcap na pasta .cuckoo/storage/analyses. A partir disso, é possível executar o arquivo dump.pcap no *Wireshark* e obter uma análise mais detalhada de como estava a rede durante a execução do *Ransomware* na VM. Na figura 3.18 pode-se observar a captura de pacotes da rede durante a análise de uma amostra do *malware*.

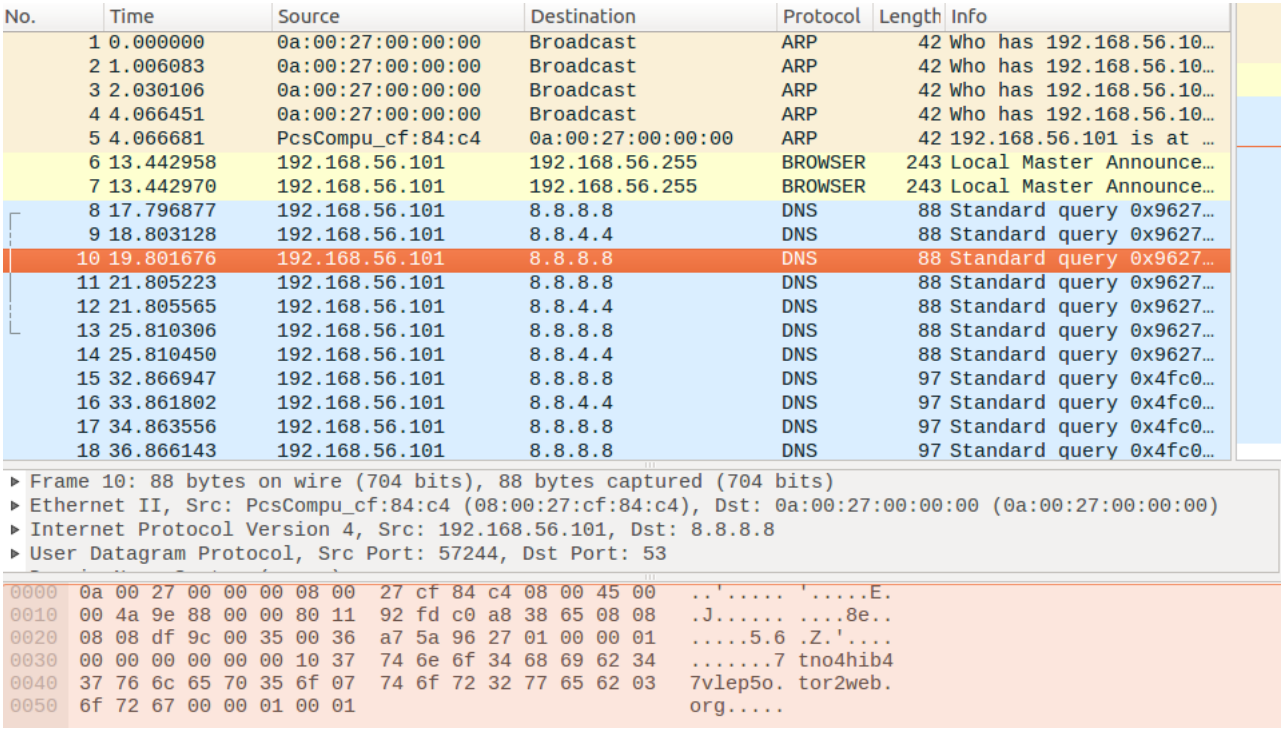

Figura 3.11: Análise de pacotes do tráfego de rede gerado durante a execução do *Ransomware*

Capítulo 4

Análises e Resultados

Neste capítulo, por meio das ferramentas apresentadas anteriormente, são apresentados todos os resultados e análises. Os relatórios gerados pela ferramenta *Cuckoo Sandbox* foram comparados, com a finalidade de se obter as características dos aspectos comportamentais dos *Ransomwares*. Foram 4 parâmetros utilizados na comparação desses *malwares*, que serão detalhados neste capítulo.

4.1 Famílias de *Ransomware* analisadas

Neste trabalho, foi utilizada apenas uma VM que executa a cada instância um arquivo de *ransomware* diferente. Seria possível a utilização de uma quantidade maior de máquinas virtuais, dependendo apenas da capacidade de processamento do computador em que essas máquinas são criadas e executadas. O *Cuckoo Sandbox* é capaz de executar e realizar os arquivos enviados para as VMs seguindo a sua política de fila. [Cuckoo , 2017]

Foram coletadas 42 amostras de *malware* para execução utilizando a metodologia que foi descrita. As amostras foram coletadas das bibliotecas de *malware* online *Malware Traffic Analysis* [Analysis , 2017], *the Zoo* [theZoo , 2017] e *Open Malware* [Malware , 2017].

Dessas, 12 amostras apresentaram alguma técnica anti-virtualização que não possibilitou a análise e ocorreu erro. Das 30 amostras restantes executadas com sucesso, foram analisados 28 arquivos de *ransomware*, 1 arquivo de *Trojan.Kovter* e 1 arquivo de *Vírus.Juegos* para base de comparação dos parâmetros com os *ransomwares*.

As amostras executadas com sucesso, acompanhadas de um identificador (ID) na ordem de que foram executados, o *hash* MD5 e SHA1 dos arquivos podem ser vistas na Tabela 4.1.

Tabela 4.1: Amostras de *malware* executadas com seus respectivos *hashes*

Malware	ID	Hashes MD5 e SHA1, respectivamente
Rex	1	5bd44a35094fe6f7794d895122ddfa62 98172e49c3d5d70ffdcefd071f9762c58430a393
Satana	2	46bfd4f1d581d7c0121d2b19a005d3df 5b063298bbd1670b4d39e1baef67f854b8dcba9d
TeslaCrypt.a	3	6e080aa085293bb9fbdcc9015337d309 51b4ef5dc9d26b7a26e214cee90598631e2eaa67
TeslaCrypt.b	4	209a288c68207d57e0ce6e60ebf60729 e654d39cd13414b5151e8cf0d8f5b166dddd45cb
TeslaCrypt.c	5	6d3d62a4cff19b4f2cc7ce9027c33be8 e906fa3d51e86a61741b3499145a114e9bfb7c56
Vipasana.a	6	2aea3b217e6a3d08ef684594192cafc8 3a0b855dd052b2cdc6453f6cbdb858c7b55762b0
Vipasana.b	7	a890e2f924dea3cb3e46a95431ffae39 35719ee58a5771156bc956bcf1b5c54ac3391593
Vipasana.c	8	adb5c262ca4f95fee36ae4b9b5d41d45 cdbe420609fec04ddf3d74297fc2320b6a8a898e
WannaCry	9	84c82835a5d21bbcf75a61706d8ab549 5ff465afaabcbf0150d1a3ab2c2e74f3a4426467
Matsnu	10	1b2d2a4b97c7c2727d571bbf9376f54f 1fc29938ec5c209ba900247d2919069b320d33b0
Petya.a	11	71b6a493388e7d0b40c83ce903bc6b04 34f917aaba5684fbe56d3c57d48ef2a1aa7cf06d
Petya.b	12	a92f13f3a1b3b39833d3cc336301b713 d1c62ac62e68875085b62fa651fb17d4d7313887
Petya.c	13	af2379cc4d607a45ac44d62135fb7015 39b6d40906c7f7f080e6befa93324dddadcbd9fa7
Radamant	14	6152709e741c4d5a5d793d35817b4c3d 05ae9c76f8f85ad2247c06d26a88bbbcfff4d62e
Cerber	15	8b6bc16fd137c09a08b02bbe1bb7d670 c69a0f6c6f809c01db92ca658fcf1b643391a2b7
CryptoLocker	16	7f9c454a2e016e533e181d53eba113bc 694dc7713537a7237030f7623881423fcb8d8c5c
Jigsaw	17	2773e3dc59472296cb0024ba7715a64e 27d99fbca067f478bb91cdbcb92f13a828b00859
Locky	18	b06d9dd17c69ed2ae75d9e40b2631b42 b606aaa402bfe4a15ef80165e964d384f25564e4
CryptoWall	19	47363b94cee907e2b8926c1be61150c7 ca963033b9a285b8cd0044df38146a932c838071
Cryptxxx	20	0d96df4667254727fed1f71d8d89318f 2d27c795ec2f0db441ad6caede6b6e285fe11897

AlphaCrypt	21	a08784f5691a0a8ce6249e1981dea82c fdfd630730da8c6dc075fb4a9a1011ec53914562
Gryphon	22	d5f8e056f470c14b36d9d42553b53197 cfa072024ffd9c2327efc3d2fd4dd90ddbd6943c
CrypMic	23	59bff0a38a04c372e4896a4fb2eea8fb 2b487ceb3b34cf509b271579160656ca6e1328e2
TeslaCrypt 2.0	24	f87893b441483020ba75c870ffb7b6af 2f622c1b053cc3244af7e75844a1d6ec0b0479c4
CryptoShield	25	cc882e0f288b8996bfa66cda9a27e137 e5686d807ada9e7e953dd2a125fdaf5be958375b
CrypFile	26	c698ded3ef2372ef943c9d0bef154275 4bcec61a6a10163d30e37cf8276fd315e4d0bd93
CHIP	27	35f68acc0c3d5761a61975ec77b49cbc f6d03e713bc9b47265141d9f9b83ae634d43d204
Jaff	28	c2a760c6461449ac1d5a5538242bed11 59684c6261afc698c0f6a46658986f0268f4c5a0
Kovter	29	26d893514b631ef1d5255f711895be09 3b4b1dc1e40758a35f5d58be8dbf57e6aa7ab619
Juegos	30	15af6227d39ca3f9d1dcd8566efb0057 c8c3bf9ed944b614ae4b3e747e69e84026fb4039

Um exemplo de análise é a execução da amostra de *malware TeslaCrypt*, que após sua execução na máquina *guest*, exibe a imagem vista na Figura 4.1 :

Figura 4.1: Tela exibida durante a execução do *Ransomware TeslaCrypt*

É possível observar que na tela exibida na Figura 4.2 uma cobrança é realizada como forma de resgate aos arquivos que foram encriptados, o que caracteriza o *TeslaCrypt* como um *Ransomware* do tipo *Crypto*.

Após a execução do malware, o relatório possui campos relativos ao que foi executado e criado pelo malware dentro da máquina virtual. Na tabela 4.2 são indicados os campos de maior relevância encontrados em um relatório gerado após a execução do Ransomware TeslaCrypt.b.

Tabela 4.2: Sumário com as principais características do relatório gerado e suas descrições

Característica	Descrição
Name	Nome do arquivo que foi executado pelo Cuckoo Sandbox
Type	Tipo do arquivo que foi executado, por exemplo: PE32 .exe for MS Windows.
Size	Tamanho do arquivo executado
SHA1	Função Hash criptográfica de 160 bits do arquivo
MD5	Função Hash criptográfica de 128 bits do arquivo
Queries for computername	O malware realiza consultas para descobrir o nome do PC que foi registrado.
Creates documents on the filesystem	Criação de documentos no sistema de arquivos da máquina alvo.
Drops a binary and executes it	Despeja um arquivo binário na máquina alvo e executa-o.
Attempts to detect Cuckoo Sandbox	O arquivo tenta detectar a presença da execução de algum arquivo que referencie a ativação do Cuckoo Sandbox.
Attempts to modify desktop wallpaper	O arquivo tenta modificar o papel de parede existente na área de trabalho.
Removes the Shadow Copy	Remove a Shadow Copy do SO para evitar a recuperação do sistema.
File has been identified by AntiVirus on ViruTotal as malicious	O arquivo executado foi identificado por diversos antivirus da plataforma VirusTotal, onde realiza a classificação primária do malware executado.
Process memory dump	Nessa parte é possível identificar as URLs que foram acessadas durante a execução do malware.

4.2 Parâmetros comportamentais analisados

Para realizar a identificação dos perfis de operação de *Ransomware*, foram selecionados 4 parâmetros comportamentais dos *malwares* que foram relevantes para extrair características de como este *malware* se comporta durante a simulação do comportamento de um arquivo que possa ser um candidato a *ransomware*. Os parâmetros selecionados foram:

- Tamanho do arquivo do *malware*;
- Ocorrência ou não da utilização de rede e os protocolos utilizados no tráfego de dados na rede;

- Criação de arquivos na máquina alvo;
- Ocorrência ou não da remoção da *Shadow Copy* do Sistema Operacional que será o alvo.

4.2.1 Tamanho do arquivo do *malware*

O tamanho de arquivo executado foi extraído das informações obtidas no relatório do *Cuckoo Sandbox* que é gerado após a execução do *malware* na VM criada. Na figura 4.2 pode-ser observar o gráfico construído a partir do tamanho de cada *malware* listado.

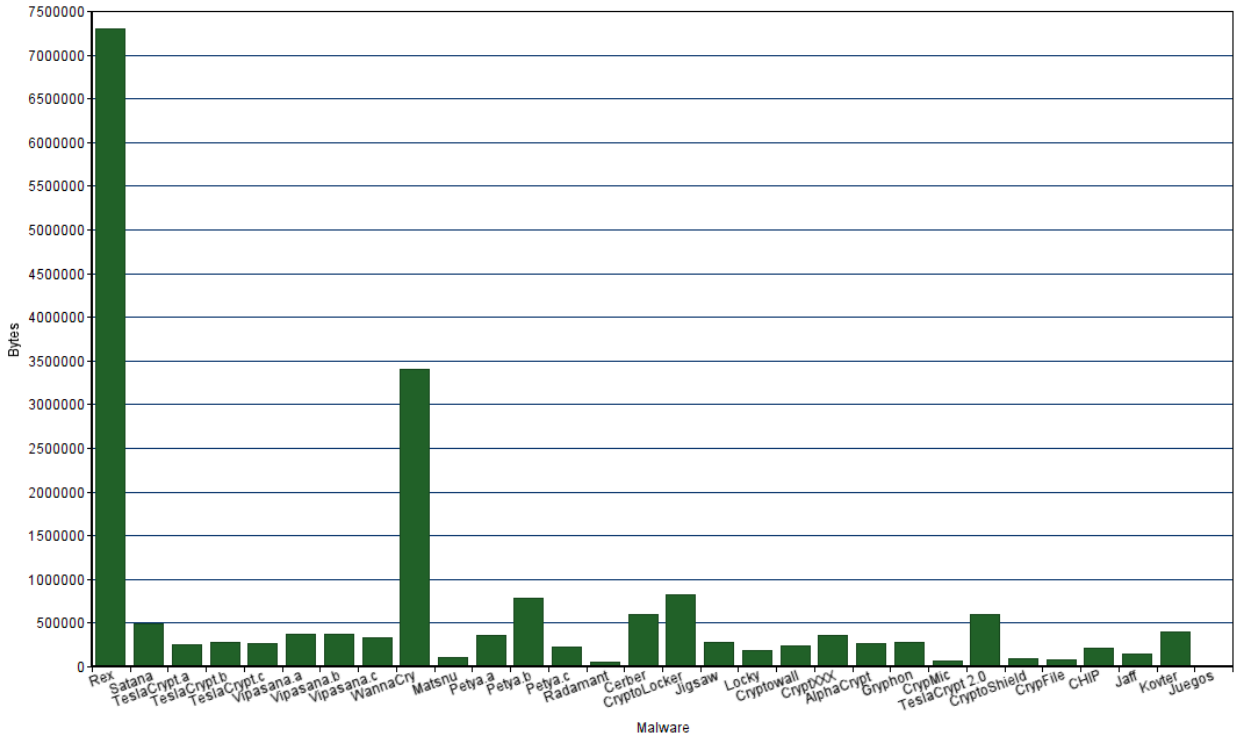

Figura 4.2: Gráfico do tamanho do arquivo do *malware*

Em relação ao quesito tamanho do arquivo, é possível notar por meio da figura 4.2, que a maiora das famílias de *Ransomwares* variam de tamanho entre 200 KB a 400 KB. Alguns "*outliers*" presentes nessa tabela possuem um tamanho mais elevado, como é o caso do *Rex Ransomware* possuindo um tamanho de 7,3 MB e do *WannaCry*, com 3,4 MB. O *Rex Ransomware*, como é um *malware* para linux, não há o que comparar em relção aos demais. Por sua vez o WannaCry pode possuir em sua estrutura arquivos com tamanhos que consomem mais espaço.

Quando comparados com outros tipos de *malware*, como o *Trojan.Kovter* e o vírus "*Juegos*", é notado que, apesar de o *Trojan.Kovter* possuir um tamanho parecido com a média de tamanhos das famílias de *Ransomwares*, o vírus está muito abaixo na escala. Isto mostra que se um arquivo suspeito com um tamanho menor que 50 KB, pode-se, com base na análise de dados desse trabalho, descartar que esse *malware* seja do tipo *Ransomware*.

4.2.2 Tráfego de dados na rede

Outro aspecto comportamental analisado foi o tráfego de dados na rede. Os protocolos requisitados nesse tráfego foram UDP, HTTP, TCP, DNS, TLS, ARP e ICMP, a quantidade de requisições com base nos 3 últimos protocolos respectivamente citados, não foram levadas em consideração a fins de análise por obter poucas requisições.

Tabela 4.3: Ransomwares que geraram tráfego de rede

Malware	ID	Tipo de Ransomware
Teslacrypt.a	3	Crypto
Teslacrypt.b	4	Crypto
TeslaCrypt.c	5	Crypto
Radamant	14	Crypto
Cerber	15	Crypto
CryptoLocker	16	Crypto
Locky	18	Crypto
CryptoWall	19	Crypto
AlphaCrypt	21	Crypto
CrypMIC	23	Crypto
TeslaCrypt 2.0	24	Crypto
CryptoShield	25	Crypto
CrypFile	26	Crypto
CHIP	27	Crypto
Jaff	28	Crypto

Alguns exemplos de requisições utilizando o protocolo UDP durante a execução do *TeslaCrypt.c* podem ser vistas na figura 4.3.

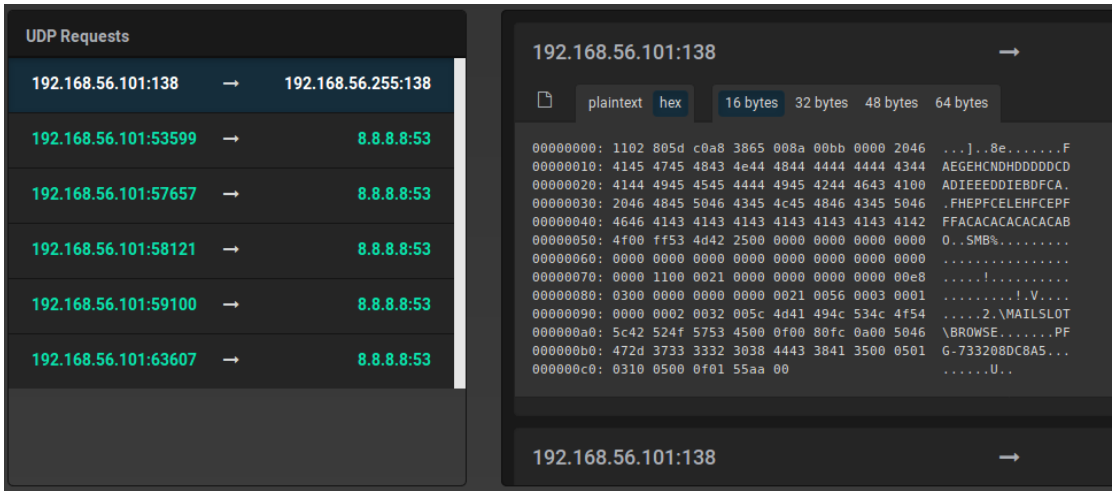

Figura 4.3: Requisições no tráfego de rede utilizando o protocolo UDP

Além disso, alguns exemplos de requisições analisadas utilizando o programa *Wireshark* dos protocolos TCP e DNS podem ser vistos nas figuras 4.4 E 4.5:

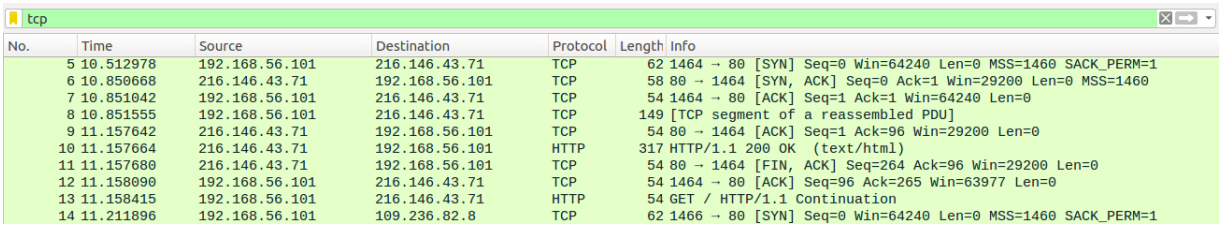

Figura 4.4: Requisições no tráfego de rede utilizando o protocolo TCP durante execução do *ransomware* CHIP

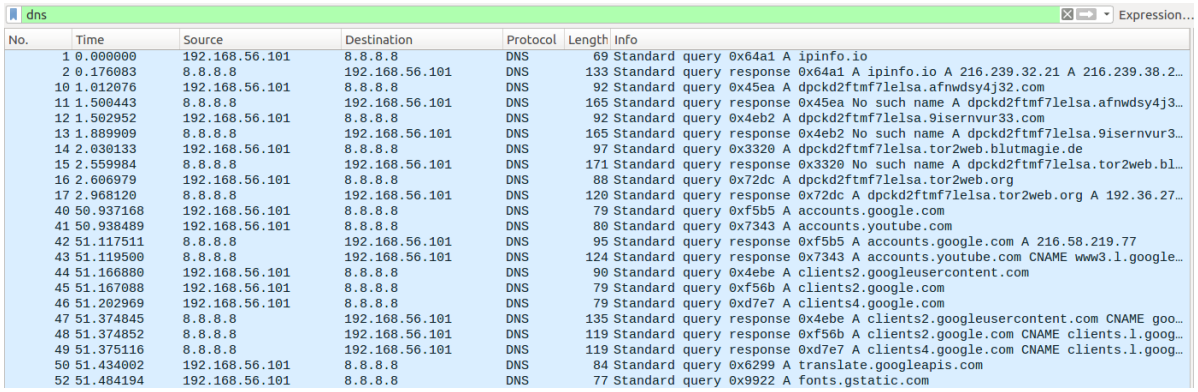

Figura 4.5: Exemplos de requisições no tráfego de rede utilizando o protocolo DNS durante a execução do *ransomware AlphaCrypt*

O único tipo de *ransomware* analisado que opera em Sistema Operacional Linux, foi o *Rex*, do tipo *Locker*. Este *malware* foi responsável por algumas tentativas de conexões na rede, porém sem sucesso, o que indica que provavelmente esse *ransomware* não foi integralmente compilado na máquina virtual, por se tratar de uma VM Windows XP SP3, ou seja, com um diferente tipo de Sistema Operacional. As requisições DNS de domínios com falhas foram:

- https://www.microsoft.com/bar.htm;
- http://www.infobeat.com;
- http://www.DocURL.com/bar.htm;
- http://shell.windows.com/fileassoc/%%04x/xml/redir.asp?Ext=%%s
- http://digitalid.verisign.com;
- http://www.microsoft.com/isapi/redir.dll?prd=ie;
- http://shell.windows.com/fileassoc/fileassoc.asp?LangID=%%04x;
- http://related.msn.com/related.asp?url

Na tabela 4.3 pode-se observar a quantidade de requisições geradas em cada protocolo relacionado a cada família de *ransomware*.

Tabela 4.4: Protocolos utilizados por cada *ransomware* no tráfego de rede e a quantidade de requisições

Malware	Tipo de Ransomware	DNS	TCP	UDP	HTTP
Teslacrypt.a	Crypto	20	12	47	2
Teslacrypt.b	Crypto	5	10	6	2
TeslaCrypt.c	Crypto	5	3	6	1
Radamant	Crypto	2	2	2	2
Cerber	Crypto	3	2	218	3
CryptoLocker	Crypto	59	0	69	0
Locky	Crypto	6	2	7	2
CryptoWall	Crypto	2	1	2	1
AlphaCrypt	Crypto	21	12	52	4
CrypMIC	Crypto	0	0	1	0
TeslaCrypt 2.0	Crypto	21	27	23	20
CryptoShield	Crypto	2	4	2	4
CrypFile	Crypto	0	1	1	1
CHIP	Crypto	2	1	2	1
Jaff	Crypto	1	0	2	0

Na tabela 4.4 é observado que todos os *Ransomwares* que fizeram o uso da rede são do tipo *Crypto*, fazendo requisições de protocolos da camada de tranposte e aplicação.

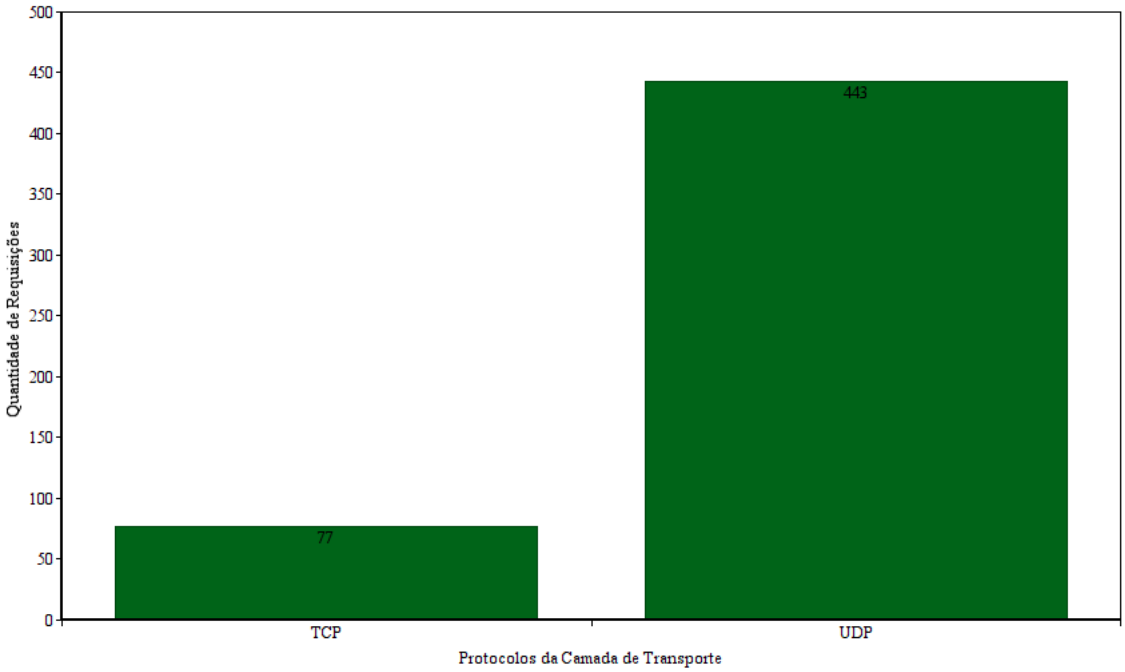

Figura 4.6: Total de requisições na camada de transporte após a execução de todos *malwares* listados

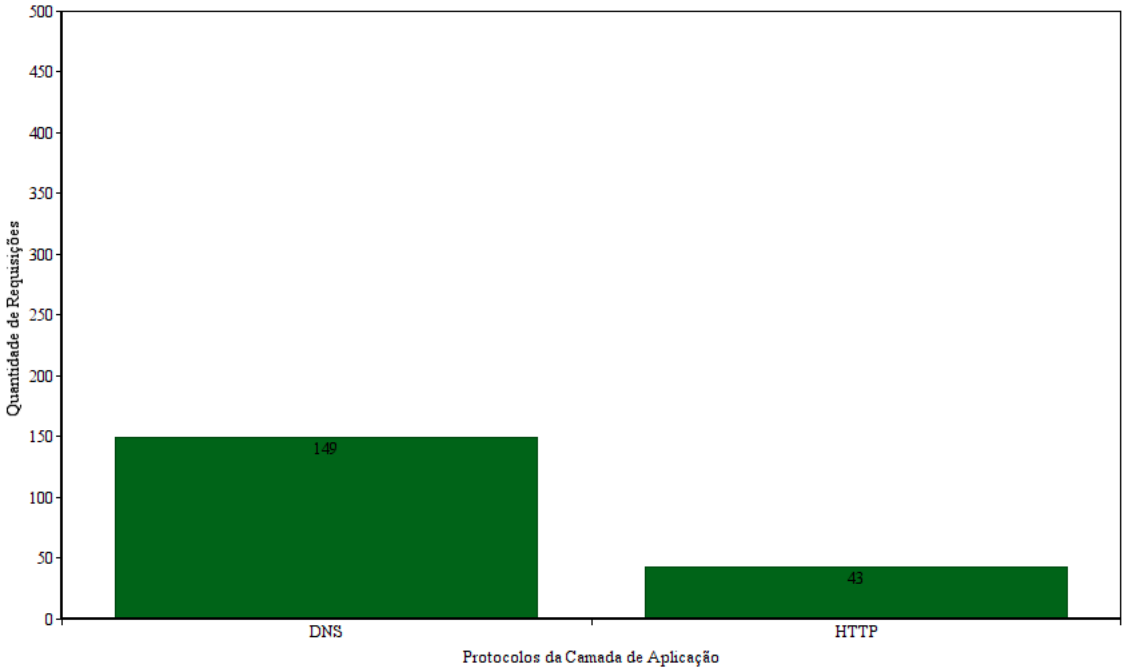

Figura 4.7: Total de requisições na camada de aplicação após a execução de todos *malwares* listados

Com base na figura 4.6, é possível perceber que, na camada de transporte, os principais protocolos envolvidos no tráfego de rede, quando um *Ransomware* é executado em determinada máquina são os protocolos TCP(*Transmission Control Protocol*) e UDP(*User Datagram Protocol*). Com destaque ao protocolo UDP, que é um protocolo não orientado para a conexão da camada de transporte do modelo TCP/IP, é um protocolo relativamente mais simples em comparação ao TCP já que não fornece controle de erros, ou seja, não está orientado para a conexão. Este protocolo foi responsável por um total de quase 450 requisições, bem a frente do segundo protocolo.

Na figura 4.7, é possível perceber que, na camada de aplicação, os principais protocolos envolvidos no tráfego de rede, quando um *ransomware* é executado em determinada máquina são os protocolos DNS (*Domain Name System*) e HTTP(*Hypertext Transfer Protocol*). O protocolo DNS foi responsável por um total de quase 150 requisições, bem a frente do HTTP, com apenas 43 requisições no total.

4.2.3 Criação de arquivos na máquina

Nesta parte é abordado um comportamento que foi percebido em 40% dos *ransomwares* analisados, ou seja, dos 28 tipos de *ransomwares* submetidos a análise da ferramenta *Cuckoo Sandbox*, 11 apresentaram a característica de criação de arquivos na máquina da vítima. Os *ransomwares* que criaram arquivos na máquina alvo podem ser vistos na tabela 4.4.

Todas as extensões dos arquivos criados foram, .doc, .docm, .docx, .xls, .ppt, .pptx, .tmp, .exe, .lnk, .pdf e .inf. De acordo com o gráfico da figura 4.7, ficam evidenciados que as principais extensões criadas foram, .doc com 23 documentos criados, seguidas de 19 documetos criados pela

Tabela 4.5: *Ransomwares* que criaram algum tipo de arquivo no máquina alvo

Malware	ID	Tipo de Ransomware
Satana	2	Crypto
Teslacrypt.a	3	Crypto
Teslacrypt.b	4	Crypto
TeslaCrypt.c	5	Crypto
WannaCry	9	Crypto
Locky	18	Crypto
AlphaCrypt	21	Crypto
CrypMIC	23	Crypto
TeslaCrypt 2.0	24	Crypto
CryptoShield	25	Crypto
CrypFile	26	Crypto

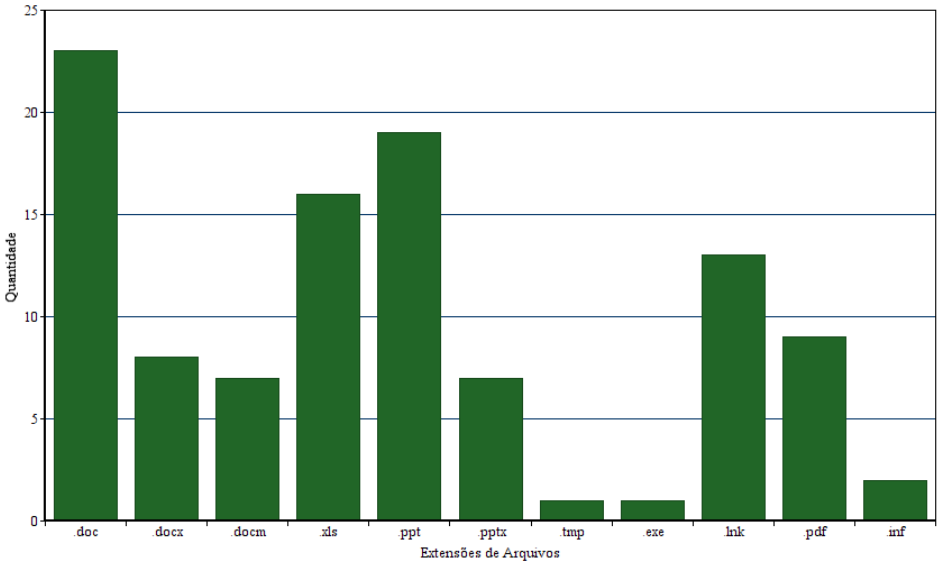

Figura 4.8: Extensões dos arquivos criados pelos *ransomwares* e suas quantidades

extensão .ppt, 16 .xls e 13 arquivos .lnk. Uma ressalva a se fazer é que somente o *WannaCry* foi responsável por todas as entensões criadas do tipo .lnk durante seu ataque.

Um ponto importante analisado é que quando se diz respeito as variações de uma mesma família de *ransomware*, quando há criação de arquivos na máquina, como é o caso do *TeslaCrypt.a*, *TeslaCrypt.b*, *TeslaCrypt.c*, *TeslaCrypt 2.0* (evolução do *TeslaCrypt*) e *AlphaCrypt* (variação do *TeslaCrypt*), todos apresentaram em comum, a criação de arquivos com mesmo nome, nas mesmas pastas, porém com *hashes* diferentes:

- `C:\\DocumentsandSettings\\DefaultUser\\Templates\\excel.xls`
- `C:\\DocumentsandSettings\\DefaultUser\\Templates\\excel4.xls;`

- `C:\\DocumentsandSettings\\DefaultUser\\Templates\\winword2.doc;`
- `C:\\DocumentsandSettings\\PFG\\Templates\\winword.doc;`
- `C:\\DocumentsandSettings\\DefaultUser\\Templates\\powerpnt.ppt.`

  Essa análise, leva ao indício de que os ransomwares citados acima e que criaram esses arquivos com mesmo nome e hashes diferentes, possuem grandes chances de terem sidos desenvolvidos e distribuídos pelo mesmo autor ou grupo de *hackers*.

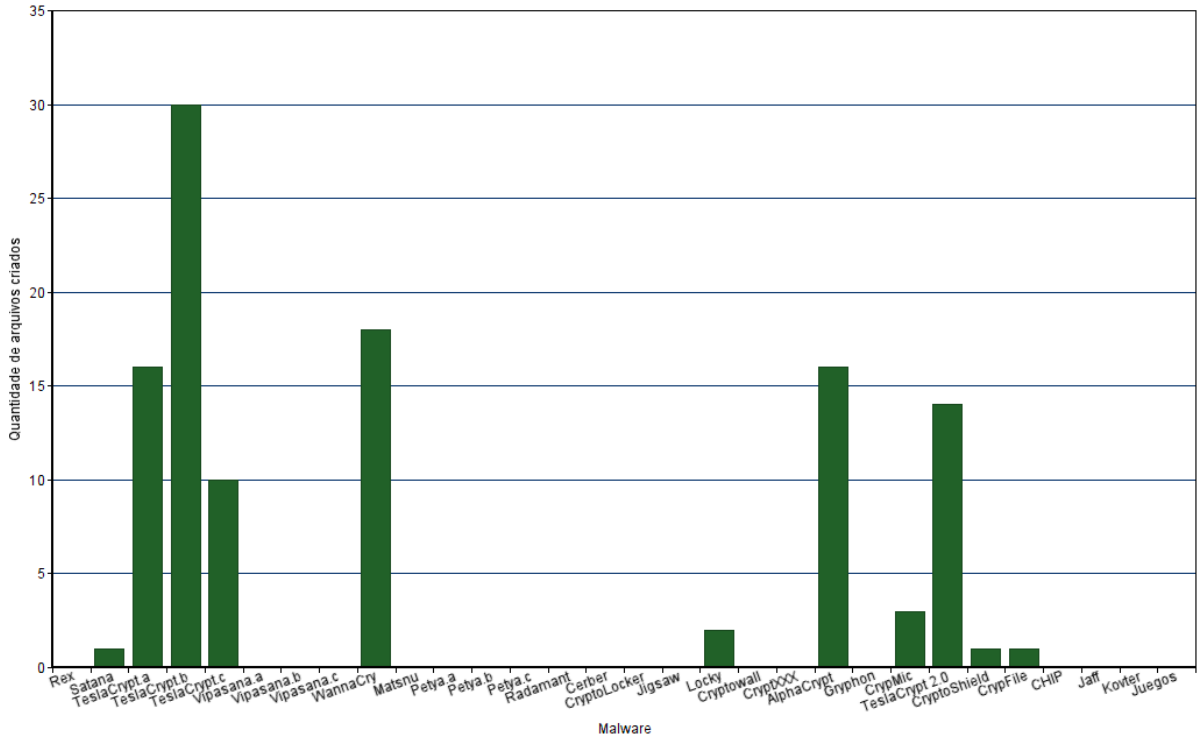

Figura 4.9: Gráfico da quantidade de arquivos criados pelo *malware* na máquina alvo

Outra característica fundamental mostrada na tabela 4.4, é que todos os *malwares* que criaram arquivos na máquina da vítima foram do tipo *Crypto*, logo, com base nas análises desse projeto, pode-se dizer que quando uma família de *ransomware* tem a característica de criar arquivos, esse *malware* provavelmente será do tipo *Crypto*.

4.2.4 Remoção da cópia sombra *Shadow Copy* do sistema

Por fim, uma característica muito importante que é possível concluir com base na análise dos dados, é diferenciar em parte o comportamento entre os *ransomwares* do tipo *Locker* e *Crypto*.

Como mostrado na tabela 4.3, pode-se observar que 100% dos *Locker ransomwares* analisados, não removem a cópia sombra do sistema (sendo que há somente *Ransomwares* do tipo Crypto na tabela), ou seja, com base no estudo de *Ransomwares* tipo *Locker*, não haveria necessidade de remover essa cópia de *backup* dos aquivos sendo que a vítima não teria acesso a esse sistema, uma

Tabela 4.6: *Malwares* que resultaram na remoção da *Shadow Copy* do sistema operacional

Malware	ID	Tipo de Ransomware
Teslacrypt.a	3	Crypto
Teslacrypt.b	4	Crypto
TeslaCrypt.c	5	Crypto
Vipasana.a	6	Crypto
Vipasana.b	7	Crypto
Vipasana.c	8	Crypto
WannaCry	9	Crypto
CryptoLocker	16	Crypto
AlphaCrypt	21	Crypto
TeslaCrypt 2.0	24	Crypto
CryptoShield	25	Crypto

vez que o *Locker Ransomware* limita o acesso do usuário a máquina, bloqueando janelas, tela, mouse e teclado, por exemplo. Por outro lado, se o *malware* remover a *Shadow Copy*, poderíamos afirmar, com base nas análises, que tem uma maior probabilidade de se tratar de um *Crypto Ransomware*.

Capítulo 5

Conclusão

Neste trabalho foi apresentado técnicas, ferramentas e estudos direcionados para classificar e comparar o comportamento e aspectos chaves das diversas famílias de *Ransomwares* presentes atualmente na rede mundial de dados, cuja característica vital é a cifragem de dados e/ou sistemas de arquivos.

Foi utilizado neste trabalho, a ferramenta *Cuckoo Sandbox*, cujo sistema de análise de *malware* avançado automatizado e código aberto permitiu o fornecimento detalhado de um relatório descrevendo o comportamento do arquivo quando executado dentro de um ambiente realista, porém isolado, ou seja, sendo executado na máquina virtual *Windows XP SP3*, usada para se obter as devidas análises.

Para tornar o trabalho completamente automatizado, uma vez que a ferramenta *Cuckoo Sandbox*, por si, já tem uma performace automática, foi desenvolvido paralelamente também um script em *Shell Script*, que é uma linguagem usada em vários sistemas operacionais, cujo objetivo foi tornar prático e fácil a execução do *malware* na VM, uma vez que por meio de uma linha de comando, a máquina virtual é ligada, e realiza o envio do arquivo macilioso, no caso do projeto, um arquivo contendo um *Ransomware*. Após a analise, é retornado um relátorio em formato HTML e apresentado para o usuário em algum navegador, tudo de forma automática, ou seja, é algo que facilita e agiliza bastante o trabalho de analistas de segurança de redes, visto que diante de um arquivo executável, é necessário apenas executar o script desenvolvido neste trabalho.

Foram analisadas ao todo, com sucesso, por meio do *Cuckoo Sandbox*, 30 *malwares*, onde 28 são *Ransomwares* de famílias variadas, sendo 4 *Ransomwares* do tipo *Locker* e o restante do tipo *Crypto*.Essa grande discrepância em relação a quantidade analisada entre os diferentes tipos de *Ransomwares* se deve pelo fato de que a grande maioria dos *Ransomwares* hoje presentes na rede mundial de dados, é do tipo *Crypto*. Contudo, mesmo com essa diferença pode-se notar caracteristicas fundamentais desse tipo de *malware*.

Dos parâmetros comportamentais observados, entre eles, tamanho do arquivo, funções *hash*, tráfego na rede, criação de arquivos na máquina e da remoção da *Shadow Copy* do sistema, pode-se concluir que, em geral, foram eficientes parar analisar os apectos dos *Ransomwares*. Os *Ransomwares*, em sua maioria possuem uma proximidade de tamanho, com alguns "*outliers*" com tamanhos

mais elevados. Porém, os *hashes* gerados pelas funções de dispersão criptográficas, MD5 e SHA1, foram ineficientes para fins de análise, pois todos os arquivos geraram *hashes* diferentes, sendo assim não há o que comparar em termos de função *hash*. É possível com base na análise do tráfego de rede, criar algum método de segurança para limitar o ataque de certos *Ransomwares*, uma vez que determinados *Ransomwares*, quando executados, requisitam o acesso a certos domínios, sendo assim é possível implementar por meio de um *proxy*, por exemplo, o bloqueio dessa requisição.

5.1 Trabalhos Futuros

Como trabalhos futuros espera-se aprodundar no tema *Ransomware*, podendo distinguir não apenas os apectos e comportamentos entre o seus tipos, *Locker* e *Crypto*, como também poder diferenciar suas diversas famílias, a partir de uma profunda análise.

Este trabalho possui uma limitação de analisar apenas 1 Ransomware por vez, ou seja, não é possível executar 2 ou mais amostras para serem analisadas ao mesmo tempo, primeiramente porque o estudo é realizado em apenas uma VM e o script foi desenvolvido apenas para efetuar a execução automática de uma amostra por vez. Esta nova ferramenta tecnológica desenvolvida, não deve ficar apenas limitada ao estudo de *Ransomwares*, devendo abranger também a pesquisa dos diversos tipos de *malwares* presentes na atualidade.

Possíveis trabalhos futuros:

- Incrementar o script para aumentar o número de amostras analisadas;
- Aumentar e refinar o rol de caracteŕisticas.
- Aumentar o número e tipo de Ransomwares analisados.

REFERÊNCIAS BIBLIOGRÁFICAS

[acunetix , 2016]ACUNETIX. *Drupal Ransomware Vulnerability Attacks – Rex.* , 2016. Acessado em Setembro de 2017. Disponível em: <https://www.acunetix.com/blog/articles/drupal-ransomware-vulnerability-attacks-rex/>.

[Analysis , 2017]ANALYSIS, M. T. *Malware Traffic Analysis.* , 2017. Acessado em Agosto de 2017. Disponível em: <http://www.malware-traffic-analysis.net/2017/index.html>.

[Arcon , 2017]ARCON. *Visibilidade é poder - Contra-atacando o Ransomware.* , 2017. Acessado em Setembro de 2017. Disponível em: <https://www.arcon.com.br/blog/visibilidade-%C3%A9-poder-contra-atacando-o-ransomware>.

[CERT.br , 2015]CERT.BR. *Cartilha de Segurança para Internet - Ransomware.* , 2015. Acessado em Julho de 2017. Disponível em: <https://cartilha.cert.br/ransomware/>.

[CNBC , 2017]CNBC. *Shipping company Maersk says June cyberattack could cost it up to $300 million.* , 2017. Acessado em Novembro de 2017. Disponível em: <https://www.cnbc.com/2017/08/16/maersk-says-notpetya-cyberattack-could-cost-300-million.html>.

[Computer , 2017]COMPUTER, B. *Locker Ransomware Information Guide and FAQ.* , 2017. Acessado em Julho de 2017. Disponível em: <https://www.bleepingcomputer.com/virus-removal/locker-ransomware-information>.

[Corporation , 2016]CORPORATION, S. *How to Remove Kollah.* , 2016. Acessado em Julho de 2017. Disponível em: <https://www.solvusoft.com/en/malware/ransomware/kollah/>.

[Cuckoo , 2017]CUCKOO, F. *Architecture - Cuckoo Sandbox.* , 2017. Acessado em Setembro de 2017. Disponível em: <http://docs.cuckoosandbox.org/en/latest/introduction/what/architecture>.

[Cuckoo , 2017]CUCKOO, F. *Cuckoo in the Background.* , 2017. Acessado em Outubro de 2017. Disponível em: <http://docs.cuckoosandbox.org/en/latest/usage/start/>.

[Cuckoo , 2017]CUCKOO, F. *Processing Modules - Cuckoo Sandbox.* , 2017. Acessado em Agosto de 2017. Disponível em: <http://docs.cuckoosandbox.org/en/latest/customization/processing/?highlight=processing>.

[Cuckoo , 2017]CUCKOO, F. *Signatures - Cuckoo Sandbox.* , 2017. Acessado em Outubro de 2017. Disponível em: <http://docs.cuckoosandbox.org/en/latest/customization/signatures/>.

[Cybereason , 2017]CYBEREASON, I. *How to defend against ransomware targeting shared network drives and cloud backups.* , 2017. Acessado em Julho de 2017. Disponível em: <https://www.cybereason.com/labs-ransomware-looks-to-strike-it-rich-by-targeting-shared-network-drives-cloud-backup-services/>.

[Demartini , 2013]DEMARTINI, F. *Windows XP é o mais infectado por vírus segundo relatório da Microsoft.* , 2013. Acessado em Julho de 2017. Disponível em: <http://www.techtudo.com.br/noticias/noticia/2013/10/windows-xp-e-o-mais-infectado-por-virus-segundo-relatorio-da-microsoft.html>.

[Digima , 2016]DIGIMA. *Malwarerid Hydracrypt - Como remover?* , 2016. Acessado em Julho de 2017. Disponível em: <http://malwarerid.com.br/malwares/hydracrypt/>.

[Digital , 2017]DIGITAL, N. *Ataques de ransomware dispositivos móveis aumentarão em 2017.* , 2017. Acessado em Outubro de 2017. Disponível em: <http://cio.com.br/noticias/2017/01/25/ataques-de-ransomware-dispositivos-moveis-aumentarao-em-2017/>.

[F-Secure , 2016]F-SECURE. *A quick guide to crypto-ransomware - what it is, how it works, what happens when your computer is infected and what you can do to protect your computer.* , 2016. Acessado em Setembro de 2017. Disponível em: <https://www.f-secure.com/en/web/labs$_global/crypto-ransomware$>.

[Forbes , 2017]FORBES. *NotPetya Ransomware Attack Cost Shipping Giant Maersk Over $200 Million.* , 2017. Acessado em Novembro de 2017. Disponível em: <https://www.forbes.com/sites/leemathews/2017/08/16/notpetya-ransomware-attack-cost-shipping-giant-maersk-over-200-million/47c023b64f9a>.

[Infowester , 2016]INFOWESTER. *O que é ransomware?* , 2016. Acessado em Julho de 2017. Disponível em: <https://www.infowester.com/ransomware.php>.

[JSON , 2017]JSON. *Introducing JSON.* , 2017. Acessado em Julho de 2017. Disponível em: <http://www.json.org/>.

[Kafeine , 2016]KAFEINE. *CryptXXX: New Ransomware From the Actors Behind Reveton, Dropping Via Angler.* , 2016. Acessado em Setembro de 2017. Disponível em: <https://www.proofpoint.com/us/threat-insight/post/cryptxxx-new-ransomware-actors-behind-reveton-dropping-angler>.

[Kaspersky , 2015]KASPERSKY. *Dez fatos sobre o ransomware.* , 2015. Acessado em Julho de 2017. Disponível em: <https://blog.kaspersky.com.br/dez-fatos-sobre-o-ransomware/4614/>.

[Kaspersky , 2016]KASPERSKY. *Decrypting CryptXXX version 3 — for free.* , 2016. Acessado em Setembro de 2017. Disponível em: <https://www.kaspersky.com/blog/cryptxxx-v3-ransomware/13628/>.

[Kaspersky , 2016]KASPERSKY. *Satana: ransomware dos infernos.* , 2016. Acessado em Outubro de 2017. Disponível em: <https://www.kaspersky.com.br/blog/satana-ransomware/6406/>.

[KrebsOn , 2012]KREBSON, S. *Inside a 'Reveton' Ransomware Operation.* , 2012. Acessado em Setembro de 2017. Disponível em: <https://krebsonsecurity.com/2012/08/inside-a-reveton-ransomware-operation/>.

[Livehacking , 2010]LIVEHACKING. *New Variant of GpCode Back – Still Demanding Ransom Money to Free Your Data.* , 2010. Acessado em Julho de 2017. Disponível em: <http://www.livehacking.com/tag/trojan-pgpcoder/>.

[Malware , 2017]MALWARE, O. *Open Malware.* , 2017. Acessado em Julho de 2017. Disponível em: <http://openmalware.org/>.

[Malwarebytes , 2017]MALWAREBYTES, L. *State of Malware Report.* , 2017. Acesso em Setembro de 2017. Disponível em: <https://www.malwarebytes.com/pdf/white-papers/stateofmalware.pdf>.

[Netsafe , 2015]NETSAFE. *Android Ransomware.* , 2015. Acessado em Julho de 2017. Disponível em: <https://www.netsafe.org.nz/dealing-with-police-ransomware-on-your-android-device/>.

[PCrisk , 2016]PCRISK. *CTB Locker - 'Your personal files are encrypted' virus removal instructions.* , 2016. Acessado em Julho de 2017. Disponível em: <https://www.pcrisk.com/removal-guides/8120-your-personal-files-are-encrypted-virus>.

[PCrisk , 2016]PCRISK. *Instruções de remoção do ransomware Ransom32.* , 2016. Acessado em Setembro de 2017. Disponível em: <https://www.pcrisk.pt/guias-de-remocao/8254-ransom32-ransomware>.

[PCrisk , 2017]PCRISK. *Ransomware CryptoShield.* , 2017. Acessado em Outubro de 2017. Disponível em: <https://www.pcrisk.pt/guias-de-remocao/8485-cryptoshield-ransomware>.

[Python , 2017]PYTHON. *Python Docs.* , 2017. Acessado em Setembro de 2017. Disponível em: <https://docs.python.org/2/>.

[Scaife]SCAIFE, . et al;. *CryptoLock (and Drop It): Stopping Ransomware Attacks on User Data.* Acessado em Julho de 2017. Disponível em: <IEEE 36th International Conference on Distributed Computing Systems, 2016>.

[Semvirus , 2016]SEMVIRUS. *Vírus ransomware Vipasana. Como remover ? (Guia de desinstalação).* , 2016. Acessado em Agosto de 2017. Disponível em: <https://semvirus.pt/virus-ransomware-vipasana/>.

[Software , 2017]SOFTWARE, E. *CHIP Ransomware.* , 2017. Acessado em Agosto de 2017. Disponível em: <https://www.enigmasoftware.com/chipransomware-removal/>.

[Symantec , 2017]SYMANTEC, C. *Internet Security Threat Report - Ransomware.* , 2017. Acesso em Outubro de 2017. Disponível em:

<https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-ransomware-2017-en.pdf>.

[theZoo , 2017]THEZOO. *Malware Samples.* , 2017. Acessado em Agosto de 2017. Disponível em: <https://github.com/ytisf/theZoo/tree/master/malwares/Binaries>.

[TrendMicro , 2017]TRENDMICRO. *Ransomware.* , 2017. Acesso em Outubro de 2017. Disponível em: <https://www.trendmicro.com/vinfo/us/security/definition/ransomware>.

[UBM , 2016]UBM. *Here Are 4 Vulnerabilities Ransomware Attacks Are Exploiting Now.* , 2016. Acessado em Agosto de 2017. Disponível em: <www.darkreading.com/vulnerabilities—threats/here-are-4-vulnerabilities-ransomware-attacks-are-exploiting-now/d/d-id/1324791>.

[VirtualBox , 2017]VIRTUALBOX. *VirtualBox User Manual.* , 2017. Acessado em Julho de 2017. Disponível em: <http://download.virtualbox.org/virtualbox/UserManual.pdf>.

[Wireshark , 2017]WIRESHARK. *Wireshark Docs.* , 2017. Acessado em Agosto de 2017. Disponível em: <https://www.wireshark.org/docs/>.

[ZEBRA , 2012]ZEBRA, N. N. *O que são máquinas virtuais?* , 2012. Acessado em Julho de 2017. Disponível em: <https://www.tecmundo.com.br/maquina-virtual/232-o-que-sao-maquinas-virtuais-.htm>.

ANEXOS

I. PRÉ-REQUISITOS PARA AS ANÁLISES

I.1 Instalação *Python 2.7*

```
sudo apt-get install git
sudo apt-get install mesa-utils
```

Para instalar o *Python 2.7*, deve-se primeiramente instalar algumas dependências para o seu funcionamento correto:

```
sudo apt-get install build-essential checkinstall
sudo apt-get install libreadline-gplv2-dev libncursesw5-dev libssl-dev libsqlite3-
dev tk-dev libgdbm-dev
```

Após a conclusão do download, verifique a pasta em que foi instalado e faça a extração do arquivo:

```
tar -xvzf Python-version.tgz
```

Em que a palavra "version" é referente à versão que o usuário realizou o download.

```
cd Python-version
```

Para finalizar a instalação, proceda com os seguintes comandos:

```
./configure
make
```

I.2 Instalação *VirtualBox*

Para configurar o repositório para o *Virtualbox*:

```
deb http://download.virtualbox.org/virtualbox/debian trusty contrib
```

Para realizar o download das dependencias da *Oracle*:

```
wget -q https://www.virtualbox.org/download/oracle_vbox_2016.asc -O- | sudo apt-key add -
wget -q https://www.virtualbox.org/download/oracle_vbox.asc -O- | sudo apt-key add -
```

O próximo passo é instalar o *Virtualbox* com os seguintes comandos:

```
sudo apt-get update
sudo apt-get install virtualbox-5.1
```

I.3 Instalação *Cuckoo Sandbox*

Para instalar corretamente o *Cuckoo Sandbox*, é necessário realizar a instalações de algumas dependências do Python:

```
sudo apt-get install python python-pip python-dev libffi-dev libssl-dev
sudo apt-get install python-virtualenv python-setuptools
sudo apt-get install libjpeg-dev zlib1g-dev swig
```

Para instalar o modelo de base de dados *opensource MongoDB*:

```
sudo apt-get install mongodb
```

Para analisar a atividade da rede realizada pelo *malware* durante a execução, é necessário um sniffer de rede configurado corretamente para capturar o tráfego e inseri-lo em um arquivo.

```
sudo apt-get install tcpdump apparmor-utils
sudo aa-disable /usr/sbin/tcpdump
sudo setcap cap_net_raw,cap_net_admin=eip /usr/sbin/tcpdump
```

É possível executar o *Cuckoo* a partir de seu próprio usuário ou criar um novo dedicado apenas para sua configuração sandbox. O usuário que executa Cuckoo deve ser o mesmo usuário que você usará para criar e executar as máquinas virtuais, caso contrário o *Cuckoo* não será capaz de identificar e executar as máquinas virtuais.

```
sudo adduser cuckoo
sudo usermod -a -G vboxusers cuckoo
```

Para instalar o *Cuckoo* no *virtualenv*:

```
virtualenv venv
. venv/bin/activate
(venv) pip install -U pip setuptools
(venv) pip install -U cuckoo
```

Alternativamente, o usuário tem a opção também de fazer o download de uma cópia do *Cuckoo Package* e instalá-lo offline, pode-se configurar o *Cuckoo* usando uma cópia em cache ou ter uma cópia de backup das versões atuais *Cuckoo*.

```
pip download cuckoo
```

O usuãrio terá um arquivo Cuckoo-2.0.0.tar.gz, ou um número maior dependendo da versão em que já estiver a plataforma do *Cuckoo Sandbox*, bem como todas as suas dependências.

```
pip install Cuckoo-2.0.0.tar.gz
pip install *.tar.gz
```

Configuração de iptables para o funcionamento do *Cuckoo* com a VM:

```
sudo iptables -t nat -A POSTROUTING -o eth0 -s 192.168.56.0/24 -j MASQUERADE
sudo iptables -P FORWARD DROP
sudo iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
sudo iptables -A FORWARD -s 192.168.56.0/24 -j ACCEPT
sudo iptables -A FORWARD -s 192.168.56.0/24 -d 192.168.56.0/24 -j ACCEPT
sudo iptables -A FORWARD -j LOG
```

No entanto, essas regras não estarão realizando qualquer encaminhamento de pacotes, a menos que o encaminhamento de IP esteja habilitado explicitamente no kernel. Para fazer isso, existe um método temporário que sobrevive até um desligamento ou reinicialização, e um método permanente que é levado em consideração ao inicializar a máquina. Simplificando, em geral, você terá que executar esses dois comandos:

```
echo 1 | sudo tee -a /proc/sys/net/ipv4/ip_forward
sudo sysctl -w net.ipv4.ip_forward=1
```

Para iniciar o cuckoo:

```
cuckoo
```

Após iniciar o Cuckoo, para iniciar a análise de um arquivo de forma manual, deve-se utilizar o comando:

```
cuckoo web runserver
```

E dessa forma o endereço 127.0.0.1:8000 estará disponível para anexar o arquivo que se deseja analisar o obter a execução do *malware* em uma máquina alvo.

I.4 Instalação *Wireshark*

```
sudo add-apt-repository ppa:wireshark-dev/stable
sudo apt-get update
sudo apt-get install wireshark
```

II. ARQUIVOS UTILIZADOS NA COMPILAÇÃO DE FERRAMENTAS

II.1 SCRIPT PARA ANÁLISE AUTOMATIZADA DO *MALWARE*

```
#!/bin/bash

#Obtido com o comando "vboxmanage list vms"
VMUUID="cd524724-5ee5-4e8c-8b9d-4df742b84bd1"

#Obtido com o comando "vboxmanage snapshot <vmuuid> list"
SNAPSHOTUUID="99a1e580-81fd-44b4-8f61-11da9a9de972"

#Porta onde a api rest do cuckoo vai rodar
APIPORT=9000

FILETOANALYSE=$1

if [ -z $FILETOANALYSE ]
then
  echo "Passe como parametro o arquivo para ser analizado"
  exit 1
fi

#Restaura o snapshot para deixar a VM utilizável pelo Cuckoo

vmRestoreState() {
  vboxmanage startvm $VMUUID --type headless
  vboxmanage controlvm $VMUUID poweroff
  vboxmanage snapshot $VMUUID restore $SNAPSHOTUUID
}

#Inicia o Cuckoo Sandbox e a API rest do cuckoo

startCuckooServer() {

  cuckoo & CUCKOOPID=$!
  export CUCKOOPID
```

```
  cuckoo api --host 127.0.0.1 --port $APIPORT & CUCKOOAPIPID=$!
  export CUCKOOAPIPID
}

#Para o Cuckoo Sandbox e a API rest do Cuckoo

stopCuckooServer() {
  kill $CUCKOOPID
  kill $CUCKOOAPIPID
}

#Parse de JSON das respostas da API rest

parsejson() {
  export PYTHONIOENCODING=utf8
  python -c "import sys, json; print (json.dumps(json.load(sys.stdin)[\"$1\"])).encode
  ('ascii')"
}

#Envia o arquivo para a API rest

sendMalware() {
  curl -F "file=@$FILETOANALYSE" http://127.0.0.1:$APIPORT/tasks/create/file
}

#Obtem o JSON de resposta da API

getResult() {
  echo $(echo $(echo $(curl http://127.0.0.1:$APIPORT/tasks/view/$1 2>/dev/null) | parsejson
  task) | parsejson status)
}

vmRestoreState
startCuckooServer
sleep 20
id=$(echo $(sendMalware) | parsejson "task_id")

while [ `getResult $id`  != '"reported"' ]
do
  sleep 20
done
curl http://127.0.0.1:$APIPORT/tasks/report/$id/html > report.html && xdg-open report.html
```

```
stopCuckooServer

exit 0
```