

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Bruno Paiva de Oliveira

**Construção de Conjuntos de Dados para
Análise de *Ransomware***

Uberlândia, Brasil

2019

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Bruno Paiva de Oliveira

Construção de Conjuntos de Dados para Análise de
Ransomware

Trabalho de conclusão de curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, Minas Gerais, como
requisito exigido parcial à obtenção do grau
de Bacharel em Ciência da Computação.

Orientador: Rodrigo Sanches Miani

Universidade Federal de Uberlândia – UFU

Faculdade de Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2019

Resumo

A cada dia são noticiados novos ataques de *ransomwares*, causando prejuízos financeiros e funcionais para empresas. Faz-se necessário que novas ferramentas de detecção sejam construídas e as já existentes, aperfeiçoadas. Para isso, o objetivo deste trabalho é disponibilizar conjuntos de dados para avaliar o desempenho dessas ferramentas e identificar possíveis indicadores de atividade maliciosa. A construção desses conjuntos foi baseada no tráfego de rede e monitoramento da atividade do computador de quatro *ransomwares*: *WannaCry*, *TeslaCrypt*, *GandCrab* e *Cerber*. Após a geração dos conjuntos, observou-se que cada *ransomware* comporta-se de uma forma diferente perante a análise de rede e os quatro possuem um comportamento similar quanto ao monitoramento do computador. Além disso, mostra-se que os conjuntos estão aptos para serem usados como objetos de entrada para algoritmos de aprendizado de máquina, podendo, assim, extrair novas estratégias para detecção do malware.

Palavras-chave: *Ransomware*, Segurança da Informação, Conjunto de Dados, Avaliação de Ferramentas, Detecção de Malware.

Lista de ilustrações

Figura 1 – Tela de resgate do <i>WannaCry</i> . Extraído de (MUNHOZ, 2017).	12
Figura 2 – Passo a passo da execução.	18
Figura 3 – Gráfico do monitoramento da memória.	21
Figura 4 – Gráfico do monitoramento do disco rígido.	22
Figura 5 – Gráfico do monitoramento do processador.	23
Figura 6 – Gráfico do monitoramento do sistema.	23
Figura 7 – Gráfico do monitoramento da memória.	25
Figura 8 – Gráfico do monitoramento do disco rígido.	25
Figura 9 – Gráfico do monitoramento do processador.	26
Figura 10 – Gráfico do monitoramento do sistema.	26
Figura 11 – Gráfico do monitoramento da memória.	28
Figura 12 – Gráfico do monitoramento do disco rígido.	28
Figura 13 – Gráfico do monitoramento do processador.	29
Figura 14 – Gráfico do monitoramento do sistema.	29
Figura 15 – Gráfico do monitoramento da memória.	31
Figura 16 – Gráfico do monitoramento do disco rígido.	31
Figura 17 – Gráfico do monitoramento do processador.	32
Figura 18 – Gráfico do monitoramento do sistema.	32
Figura 19 – Parte de um arquivo de fluxo rotulado.	33
Figura 20 – Parte de um arquivo de desempenho do computador rotulado.	34
Figura 21 – Resultado da fase de teste do algoritmo Random Forest.	36
Figura 22 – Variáveis com maior importância na geração da árvore.	36
Figura 23 – Árvore de decisão gerada usando <i>Random Forest</i>	37
Figura 24 – Resultado da fase de teste do algoritmo KNN.	38

Lista de tabelas

Tabela 1	–	Tabela com a descrição do tráfego de rede do <i>WannaCry</i>	20
Tabela 2	–	Tabela comparando quantidade de cada tipo de fluxo do <i>WannaCry</i> . .	20
Tabela 3	–	Tabela com a descrição do tráfego de rede do <i>TeslaCrypt</i>	24
Tabela 4	–	Tabela comparando quantidade de cada tipo de fluxo do <i>TeslaCrypt</i> . .	24
Tabela 5	–	Tabela com a descrição do tráfego de rede do <i>GandCrab</i>	27
Tabela 6	–	Tabela comparando a quantidade de cada tipo de fluxo do <i>GandCrab</i> .	27
Tabela 7	–	Tabela com a descrição do tráfego de rede do <i>Cerber</i>	30
Tabela 8	–	Tabela comparando a quantidade de cada tipo de fluxo do <i>Cerber</i> . . .	30
Tabela 9	–	Tabela com as descrições dos atributos gerados pelo Argus	33
Tabela 10	–	Tabela com a quantidade de métricas em cada tipo de monitoramento do computador	34

Lista de abreviaturas e siglas

AES	Advanced Encryption Standard
API	Application Programming Interface
ARP	Address Resolution Protocol
C&C	Command and Control
CSV	Comma-separated values
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
IP	Internet Protocol
KNN	K-nearest Neighbor
RaaS	Ransomware as a Service
RDS	Redes Definidas por Software
SDI	Sistemas de Detecção de Intrusão
SI	Segurança da Informação
SMB	Server Message Block
SSH	Secure Shell
TCP	Transmission Control Protocol
TI	Tecnologia da Informação
TOR	The Onion Router

Sumário

1	INTRODUÇÃO	7
2	FUNDAMENTAÇÃO TEÓRICA	10
2.1	Segurança da informação	10
2.2	<i>Ransomware</i>	11
2.3	Detecção de <i>malware</i>	13
2.4	Trabalhos Relacionados	14
3	DESENVOLVIMENTO	17
3.1	Estratégias para a construção do conjunto de dados	17
3.2	Ferramentas utilizadas e desenvolvidas	19
4	RESULTADOS	20
4.1	Experimento 1 - <i>WannaCry</i>	20
4.1.1	Tráfego de rede	20
4.1.2	Monitoramento do computador	21
4.2	Experimento 2 - <i>TeslaCrypt</i>	24
4.2.1	Tráfego de rede	24
4.2.2	Monitoramento do computador	24
4.3	Experimento 3 - <i>GandCrab</i>	27
4.3.1	Tráfego de rede	27
4.3.2	Monitoramento do computador	27
4.4	Experimento 4 - <i>Cerber</i>	30
4.4.1	Tráfego de rede	30
4.4.2	Monitoramento do computador	30
4.5	Arquivos rotulados	33
4.6	Classificação do tráfego usando aprendizado de máquina - Exemplo de aplicação do trabalho	35
5	CONCLUSÃO	39
	REFERÊNCIAS	40

1 Introdução

A evolução da tecnologia, e principalmente da Internet, revolucionou a forma como empresas conduzem seus processos (DHILLON; BACKHOUSE, 2000). Existem diversas formas de identificar a dependência direta de Tecnologia da Informação (TI) das grandes empresas (NORDSTRÖM; SÖDERSTRÖM; HANSETH, 2000): base de dados de clientes, comunicação entre funcionários e a gestão financeira são exemplos de como as empresas usam a TI para seu benefício.

Contudo, essas organizações estão sob diversos riscos referentes à Segurança da Informação (SI), uma vez que podem ocasionar consequências financeiras e de credibilidade. Por isso, muitas empresas estão colocando a SI como prioridade, alocando recursos e tempo para garantir a proteção dos seus dados (BULGURCU; CAVUSOGLU; BENBASAT, 2010).

De acordo com um relatório feito pela *Accenture* (Ponemon Institute LLC, Accenture, 2019), em 2018, o custo de crimes cibernéticos por empresa foi de 13 milhões de dólares, um aumento de 12% se comparado com 2017. A pesquisa também ressalta que os setores bancário e de utilidades são os que possuem maior custo, com 18.3 e 17.8 milhões de dólares respectivamente.

Malwares são softwares maliciosos com o objetivo de se infiltrar em um sistema computacional e causar danos ou roubo de informações, sendo uma das principais ameaças de segurança enfrentadas pela comunidade de TI (KHARAZ et al., 2016). *Ransomware* é um tipo de *malware* que se tornou o centro das atenções desde 2016, com uma alta capacidade de se autopropagar e possibilidades de interromper o funcionamento padrão da Internet (DU, 2016). A *Accenture*, no mesmo relatório, cita que o *Ransomware* foi um dos ataques que mais cresceram em 2018, possuindo um custo de aproximadamente 645 milhões de dólares para organizações. Essa ameaça é capaz de bloquear o acesso a praticamente todos os arquivos da vítima e liberar o acesso somente após o pagamento de uma quantia, que normalmente é feita através de criptomoedas. Existem basicamente dois tipos principais de *Ransomware*: *crypto* e *locker-based* (CUSACK; MICHEL; KELLER, 2018). O primeiro criptografa os arquivos do usuário, mas não bloqueia o acesso ao computador. Já o segundo, apenas impossibilita a vítima de utilizar a máquina.

Diversas técnicas e ferramentas estão sendo utilizadas para fazer a detecção de *ransomware*, entre elas Antivírus, Redes Definidas por Software (RDS) e Sistemas de Detecção de Intrusão (SDI). Contudo, são necessários estudos para o desenvolvimento de novas formas de detecção, já que na primeira metade de 2018 foram descobertas 118 novas famílias de *ransomwares* (TREND LABS, 2018).

A detecção do *ransomware* tem se tornado um desafio para a comunidade de TI, uma vez que esse *malware* ainda continua causando danos consideráveis. De acordo com [Scaife et al. \(2016\)](#), a razão para esse desafio é que os *ransomwares* realizam atividades no sistema de forma semelhante ao de programas legítimos e também utilizam como alvos usuários leigos que não possuem devida proteção no sistema, como manter os programas atualizados por exemplo. Outro ponto discutido é a facilidade de se criar um *ransomware* ou até mesmo de obter um na Internet.

Os softwares e sistemas de rede estão ficando cada vez mais complexos, e portanto, desenvolvê-los sem falhas se torna uma tarefa difícil ([GUPTA et al., 2009](#)). Em conjunto com esse desenvolvimento “falho”, criadores de códigos maliciosos também estão produzindo *malwares* cada vez mais sofisticados e robustos ([GUPTA et al., 2009](#)). Portanto, a contramedida nesse caso é o desenvolvimento de ferramentas melhores e atualizadas para detectar essa constante mudança dos *malwares*. Uma forma de avaliar a qualidade dessas ferramentas é através de métricas de comparação entre elas, ou seja, para um mesmo ambiente de testes simular o comportamento de cada uma delas e analisar os resultados. Assim, é possível determinar quais realmente detectaram a ameaça, quais foram as taxas de falso positivo e falso negativo, e, por fim, a eficiência na detecção.

A detecção do *ransomware* deve ser feita o mais rápido possível para evitar uma grande perda de dados que podem ser cruciais para uma empresa ou usuário comum. Portanto, é de suma importância que ferramentas estejam sempre atualizadas, já que novas famílias do *malware* são lançadas constantemente.

Conjuntos de dados devidamente rotulados de acordo com o problema são formas poderosas de realizar comparações de algoritmos e ferramentas ([VERBERT et al., 2012](#)). A importância de se ter conjuntos de dados disponibilizados para a comunidade é tão grande que eles já estão se tornando resultados de várias pesquisas e trabalhos ([DEKKER, 2006](#)). Porém, é notório que a disponibilização de conjuntos associados a problemas em segurança da informação ainda é precária. [Ring et al. \(2019\)](#) discutem essa questão e comentam que existe uma falta da divulgação de conjuntos de dados públicos nesse contexto.

O objetivo deste trabalho é disponibilizar conjuntos de dados para impulsionar a análise e desenvolvimento de ferramentas ou técnicas de detecção de *ransomware*. Tais conjuntos contém o tráfego de rede gerado por quatro diferentes famílias do *malware*, juntamente com o monitoramento das atividades do computador, como o uso de memória, disco rígido, processador e sistema.

Nos conjuntos gerados neste trabalho, foi considerado de suma importância que os dados fossem realísticos e simulassem ao máximo o comportamento de um usuário comum navegando na Internet. Portanto, a execução dos *ransomwares* se deu quando websites estavam sendo pesquisados e carregados. Logo, obteve-se dados tanto maliciosos

quanto normais, devidamente rotulados em um período de quinze minutos. *WannaCry*, *Cerber*, *GandCrab* e *TeslaCrypt* foram as famílias utilizadas para gerar os conjuntos, uma vez que elas apresentaram um grande impacto operacional em empresas. Na China, o *WannaCry* afetou cerca de 30 mil organizações (SOO; NG; CHEN, 2017). Já o *Cerber* afetou milhões de usuários do Microsoft Office 365 (OLENICK, 2016). *TeslaCrypt* em 4 meses conseguiu obter 76.522 dólares através das infecções (KEANE, 2015). Um hospital em Massachusetts foi infectado pelo *GandCrab* e teve dados de pessoais de pacientes, endereços e outras informações valiosas criptografadas (OLENICK, 2019).

Para construir os conjuntos de dados do tráfego de rede, foi utilizado o software Wireshark¹, responsável por capturar os pacotes trafegados e o Argus² para converter a saída do Wireshark em fluxos. Já para os conjuntos de monitoramento do uso do computador, foi utilizado o Monitor de Sistemas do Windows, que já é uma ferramenta embarcada do SO.

Este trabalho é dividido da seguinte maneira. No Capítulo 2 é discutida a fundamentação teórica, abordando temas importantes para o entendimento dessa pesquisa e abrangendo trabalhos relacionados. Já no Capítulo 3 são descritos os experimentos realizados e o processo de criação dos conjuntos. O Capítulo 4 apresenta os resultados obtidos com os experimentos. Por fim, o Capítulo 5 apresenta as conclusões e trabalhos futuros.

¹ <<https://www.wireshark.org/>>

² <<https://openargus.org/>>

2 Fundamentação Teórica

Neste capítulo são apresentados os conceitos básicos necessários para compreensão deste trabalho bem como os trabalhos relacionados.

2.1 Segurança da informação

Definido pela (ABNT, 2005), o termo segurança da informação diz respeito a capacidade das organizações de proteger seus ativos, ou seja, tudo que tenha valor, sensibilidade e criticidade. Para cada ativo, também é definido pela norma que sejam assegurados três princípios básicos: confidencialidade, integridade e disponibilidade.

Pode-se definir a tríade acima a partir do conceito proposto em (FIPS, 2004):

- Confidencialidade: restringir acesso a informações privadas somente para quem possui autorização e garantir a privacidade e controle de informações privadas.
- Integridade: impedir que dados sejam modificados ou destruídos de forma imprópria, ou seja, sem autorização.
- Disponibilidade: garantir que o acesso esteja sempre disponível e confiável.

Contudo, de acordo com Dhillon e Backhouse (2000), os três princípios acima são importantes mas possuem suas limitações, uma vez que é importante considerar também não só os dados a serem protegidos, mas como também sua interpretação. Um caso citado por Dhillon e Backhouse (2000) comenta que em 1994 um incidente de fraude no Departamento de Segurança Social de Londres acarretou 3 milhões de dólares em prejuízo. Nesse incidente, nenhum dos três pilares acima havia sido quebrado, o problema foi a interpretação.

Dhillon e Backhouse (2000) sugerem que quatro novos princípios sejam seguidos:

- Responsabilidade: todos os funcionários de um organização devem saber exatamente quais são seus deveres e responsabilidades.
- Integridade dos funcionários: como a informação é fundamental para empresa, seu vazamento pode prejudicá-la em prol dos concorrentes. Um funcionário mal intencionado pode vazar a informação sem removê-la (o que não infringe o princípio da integridade do (FIPS, 2004)).

- Confiabilidade: os trabalhadores devem ser capazes de confiar um nos outros e aceitar as regras da empresa. Em casos onde as equipes de trabalho não possuem muitos encontros físicos, é necessário se encontrarem em uma certa frequência para que os valores de confiança possam ser restabelecidos.
- Ética: não é possível que a empresa crie regras para todos os diversos possíveis acontecimentos, portanto é esperado que os funcionários agem de acordo com ética.

A segurança da informação deveria ser uma das prioridades mais altas de uma organização, já que seu comprometimento pode acarretar em danos financeiros e na credibilidade da empresa (CAVUSOGLU et al., 2004). Geralmente, as empresas baseiam sua segurança em torno de produtos tecnológicos para essa finalidade, tais como antivírus e *firewalls* por exemplo, contudo é preciso também focar em questões sócio organizacionais. No trabalho proposto em (BULGURCU; CAVUSOGLU; BENBASAT, 2010), o autor comenta a necessidade de mover a questão de segurança da informação também para o lado dos funcionários, uma vez que eles são um ponto de ruptura que muitas vezes é explorado em um ataque. É possível notar que essa visão está de acordo com os quatro novos princípios citados por Dhillon e Backhouse (2000).

2.2 Ransomware

A palavra *ransomware* vem da junção de “ransom”, “resgate” em português, com “malware”. De acordo com Gazet (2010), *ransomware* é um tipo de *malware* que demanda um pagamento em troca de uma funcionalidade roubada. Entende-se funcionalidade roubada como alguma restrição imposta pelo *malware* sobre o computador, afetando a experiência do usuário. Algumas das diversas famílias desse *malware* inativam acesso ao computador ou utilizam de técnicas avançadas para criptografar arquivos da vítima. A maioria dos *ransomwares* atuam de forma semelhante, seguindo três passos básicos:

1. Busca do alvo: normalmente o *malware* foca em arquivos que possam conter dados importantes para a vítima, como documentos de texto e fotos, uma vez que criptografar todo o disco rígido consumiria muito tempo.
2. Extorsão: nessa etapa a vítima perde acesso a alguns dos seus dados que foram bloqueados no passo anterior.
3. Exibe uma mensagem de resgate: é mostrado à vítima que ela só terá acesso de volta aos dados mediante um pagamento.

De acordo com Stallings (2013), os ataques à segurança podem ser divididos em ataques ativos e passivos. Os ativos possuem o objetivo de alterar recursos do sistema,

podendo ou não afetar sua operação. Já os passivos são aqueles em que a finalidade é descobrir ou utilizar informações do sistema, apenas monitorando e não afetando seu desempenho. O *ransomware* é classificado como um ataque ativo, já que compromete o acesso aos dados ou ao próprio sistema.

Um exemplo de mensagem de resgate exibido pelo *ransomware* *WannaCry* pode ser visto na Figura 1.



Figura 1 – Tela de resgate do *WannaCry*. Extraído de (MUNHOZ, 2017).

O *ransomware* do tipo *crypto* utiliza da criptografia para cumprir seu objetivo. A criptografia simétrica, que usa a mesma chave de criptografia tanto no atacante quanto na vítima, foi utilizada nas primeiras aparições do *malware* e foi facilmente quebrada por pesquisadores. Apesar de ser um método rápido, era falho e foi rapidamente substituído pela versão assimétrica. Utilizando essa outra técnica, o atacante utilizava duas chaves, uma pública e outra privada, onde essa ficava guardada e era usada somente após a vítima pagar o resgate. Atualmente, as famílias mais recentes do *ransomware* usam uma mistura entre cifragem simétrica e assimétrica para criptografar os dados da vítima (AKBANOV et al., 2018).

Em meados da década de 90, Young e Yung (1996) já comentavam sobre softwares maliciosos capazes de impedir a vítima de acessar seus dados, quebrando o princípio da disponibilidade e integridade. Isso mostra que essa classe de software é antigo e sua história é longa. Richardson e North (2017) citam uma evolução do *ransomware* de acordo com o passar dos anos. Alguns exemplos desse *malware* são:

- AIDS Trojan que, em 1989, foi disseminado em uma conferência da Organização Mundial da Saúde. O *malware* criptografava arquivos da vítima.
- GPCoder foi espalhado via email em 2005 e usava a criptografia simétrica.

- Locker, onde foi visto pela primeira vez na Rússia. O objetivo era bloquear o computador mostrando uma imagem pornográfica.

2.3 Detecção de *malware*

Muitas pesquisas e trabalhos estão sendo desenvolvidos na área de detecção de *malware*, uma vez que esses representam uma grande ameaça para usuários e empresas (KOLBITSCH et al., 2009).

Alguns modelos baseados em rede estão sendo utilizados, principalmente os Sistemas de Detecção de Intrusão (SDI). Os SDI's utilizam do tráfego de rede para detectar atividades maliciosas, porém somente quando há evidências de que um ataque ou intrusão está ocorrendo é que seus mecanismos são ativados (FRANCO; PAGANI, 2016). Normalmente, tais sistemas são divididos em três categorias: assinatura, especificação e anomalia. O primeiro age baseado em uma base de assinaturas contendo comportamentos suspeitos e maliciosos. Caso algum tráfego acione algum padrão estabelecido nessa base, alertas são gerados para o administrador de rede. Os SDI's por especificação focam na execução correta e especificada de um programa ou protocolo, monitorando sua atividade e alertando caso algo não estipulado ocorra. Por fim, aqueles por anomalia agem de acordo com o comportamento habitual da rede, reconhecendo padrões a partir de aprendizado de máquina (VIEGAS, 2016) e identificando possíveis intrusos quando algo fora do comum (anomalia) ocorre (FRANCO; PAGANI, 2016). Contudo, *malwares* que não produzem tráfego de rede não serão capturados pelos modelos de detecção por rede. E aqueles que produzem podem evadir a detecção simulando e combinando-se com tráfego normal (KOLBITSCH et al., 2009).

Uma outra forma de detecção de *malware* é através de estratégias *host-based* (baseado no hospedeiro, ou seja, na própria máquina). Um dos exemplos mais comumente encontrados nessa categoria são os antivírus. A análise estática é uma técnica utilizada no *host-based* e possui a capacidade de identificar um *malware* antes mesmo de ele ser executado, realizando engenharia reversa para analisar o código do programa. Os antivírus fazem a análise estática através de assinaturas de *hashes* e bytes, comparando um possível arquivo malicioso com essa base de assinaturas. Contudo, técnicas de polimorfismo e obscuração são frequentemente utilizadas pelos *malwares* para evadir a detecção, dificultando a tarefa da análise estática. Em contrapartida, existe a análise dinâmica, que mapeia comportamentos suspeitos e maliciosos durante a execução de um programa. Rastreamento de chamadas de sistemas, fluxos de controle, parâmetros e instruções de funções são exemplos do que é utilizado nesse tipo de análise. Porém, uma das formas na qual *malwares* tentam evadir a detecção é através de reordenação de chamadas de funções por exemplo, enganando o padrão estabelecido pela análise dinâmica (TAHIR, 2018).

2.4 Trabalhos Relacionados

Nessa seção são apresentados alguns trabalhos relacionados com o tema desta monografia.

No trabalho proposto em (KHARAZ et al., 2016), uma técnica de detecção de *ransomware* chamada *Unveil* foi utilizada. A pesquisa comenta o fato de que muitos softwares anti-*malware* não são capazes de detectar um *ransomware* de forma eficaz, ou seja, antes da vítima perder dados ou o acesso ao computador. A técnica simula um ambiente para tentar detectar a atividade do *ransomware* antes que ele seja capaz de criptografar os dados reais da vítima ou bloquear o uso do sistema. Inclusive, durante as pesquisas e desenvolvimento do trabalho, foi descoberto uma nova família de *ransomware* que não havia sido documentada pela comunidade ainda. Em famílias onde o *malware* criptografa arquivos, o *Unveil* não obteve nenhuma taxa de falso negativo. Contudo, o autor comenta sobre as limitações da técnica e que como os softwares maliciosos sempre tendem a adotar novas estratégias evasivas, é preciso cautela. Um exemplo citado é o caso do *ransomware* que atua no nível do *kernel* do sistema operacional. Esse tipo de ameaça pode desativar algumas funcionalidades de monitoramento do *Unveil* e burlar a detecção.

Devido a questões de privacidade, muitos conjuntos de dados não são disponibilizados e isso afeta a avaliação de desempenho dos Sistemas de Detecção de Intrusão. (SHIRAVI et al., 2012) descreve o processo de geração de um conjunto de dados com atividades maliciosas e normais, tentando simular ao máximo um comportamento dinâmico semelhante a situações reais. O conjunto disponibilizado contém tráfego de:

- HTTP
- SMTP
- SSH
- IMAP
- POP3
- FTP

Ataques foram simulados e o conjunto foi dividido em sete partes, cada parte correspondendo a um dia e possuindo as seguintes especificações:

1. Sexta-feira, 11/06/2010 - Tráfego Normal
2. Sábado, 12/06/2010 - Tráfego Normal
3. Domingo, 13/06/2010 - Tráfego Normal + infiltração interna

4. Segunda-feira, 14/06/2010 - Tráfego Normal + HTTP DoS
5. Terça-feira, 15/06/2010 - DoS Distribuído utilizando IRC Botnet
6. Quarta-feira, 16/06/2010 - Tráfego Normal
7. Quinta-feira, 17/06/2010 - Tráfego Normal + Força Bruta SSH

O autor comenta que sempre haverá pontos negativos e falhas em um conjunto de dados e que um conjunto perfeito nunca existirá, onde a melhor forma de geração é através de técnicas dinâmicas onde seja possível modificar, estender e reproduzir os conjuntos.

[Sharafaldin, Lashkari e Ghorbani \(2018\)](#) também comentam sobre a dificuldade de encontrar conjuntos de dados válidos e compreensíveis. Ele cita que muitos conjuntos precisam ser anonimizados devido a questões de privacidade e segurança e isso acarreta na perda de qualidade. Os autores disponibilizam um conjunto com ataques atualizados de:

- DoS
- DDoS
- Brute Force
- XSS
- SQL Injection
- Infiltration
- Port Scan
- Botnet

Para gerar esse conjunto, foi criado um ambiente para a vítima e outro para o atacante. Diferentemente do que comumente é utilizado na geração de conjuntos de dados, o ambiente da vítima contém todos os equipamentos necessários e comuns, tais como: switches, roteadores, firewalls e diferentes sistemas operacionais. O autor se preocupou muito em relação a qualidade do tráfego benigno de segundo plano, utilizando dados de 25 usuários baseados em HTTP, HTTPS, FTP, SSH e protocolos de email.

O conjunto de dados gerado contém tráfego de 5 dias consecutivos onde diversos ataques foram executados em cada dia, semelhante ao trabalho proposto em ([SHIRAVI et al., 2012](#)).

Após gerar o tráfego, [Sharafaldin, Lashkari e Ghorbani \(2018\)](#) utilizaram um algoritmo de floresta randômica para separar as melhores características de cada tipo de

ataque. No caso do ataque força bruta, as seguintes características *Initial Windows Bytes* e *flags* de *Synchronization* (SYN), *Acknowledge* (ACK) e *Push* (Psh) foram consideradas mais relevantes para detectar tal tipo de ataque. Por fim, sete algoritmos distintos de aprendizado de máquina foram utilizados para avaliar a qualidade do conjunto de dados gerado.

Akbanov et al. (2018), em seu trabalho sobre análise estática e dinâmica do *ransomware WannaCry*, comentam sobre a importância da criação de novas medidas de segurança além das tradicionais, uma vez que os *ransomwares* estão em constante evolução.

Na análise estática, usando o software Pestudio, Akbanov et al. (2018) relatam que provavelmente o *WannaCry* utilizou as bibliotecas *msvcrt.dll* e *kernel32.dll* para implementar sua criptografia. Outro ponto observado pelo Pestudio é que as chaves de criptografias simétricas e assimétricas são gerados através de chamadas na API (Application Programming Interface) *crypto* da Microsoft. Comunicações com as API's de servidor de arquivo e *runtime C* também são feitas.

Já na análise dinâmica, foi montado um ambiente com máquinas virtuais para acompanhar a execução do *malware*. Foi observado que o *ransomware* apaga todas as cópias do *Shadow Copies* do Windows, tenta impedir que a vítima entre em modo de segurança, cria um registro com a finalidade de que seja executado toda vez que a máquina for reiniciada, entre outros, garantindo a persistência e ofuscação do *ransomware*. Voltando para o tráfego de rede, notou-se que o *malware* tenta se propagar tanto na rede interna quanto externa através da exploração do SMB na porta 445 do protocolo TCP. Para comunicação com os servidores C&C (*Command and Control*), os autores identificaram endereços da rede TOR que eram chamados através das portas 9000 e 9050.

3 Desenvolvimento

Este capítulo tem o objetivo de apresentar o processo de geração dos dados e quais ferramentas foram utilizadas para a criação do conjuntos.

3.1 Estratégias para a construção do conjunto de dados

Para a construção dos conjuntos de dados foi montado um ambiente seguro com uma máquina virtual Windows 7 e executado os seguintes passos:

1. Início do monitoramento do tráfego de rede e performance do sistema.
2. Início do tráfego normal, visitando o site “outlook.com” por cerca de dois minutos.
3. Início da execução do *ransomware* e mudança no tráfego para o site “facebook.com” por cerca de quatro minutos.
4. Mudança no tráfego para o site “g1.com.br” por cerca de cinco minutos.
5. Mudança no tráfego para o site “youtube.com” por cerca de quatro minutos.
6. Finalizada captura do tráfego de rede e performance do computador, totalizando quinze minutos de dados.
7. Tratativa e conversão dos dados obtidos.
 - a) Conversão do tráfego de rede para fluxos e posteriormente para *CSV*¹.
 - b) Conversão da performance do sistema para arquivo *CSV*.
8. Rotulação dos arquivos *CSV*, onde para cada linha de monitoramento foi marcado se esse é malicioso ou normal.

O processo acima é ilustrado na Figura 2:

¹ *Csv* é um formato de arquivo de texto onde os valores são separados por vírgulas.

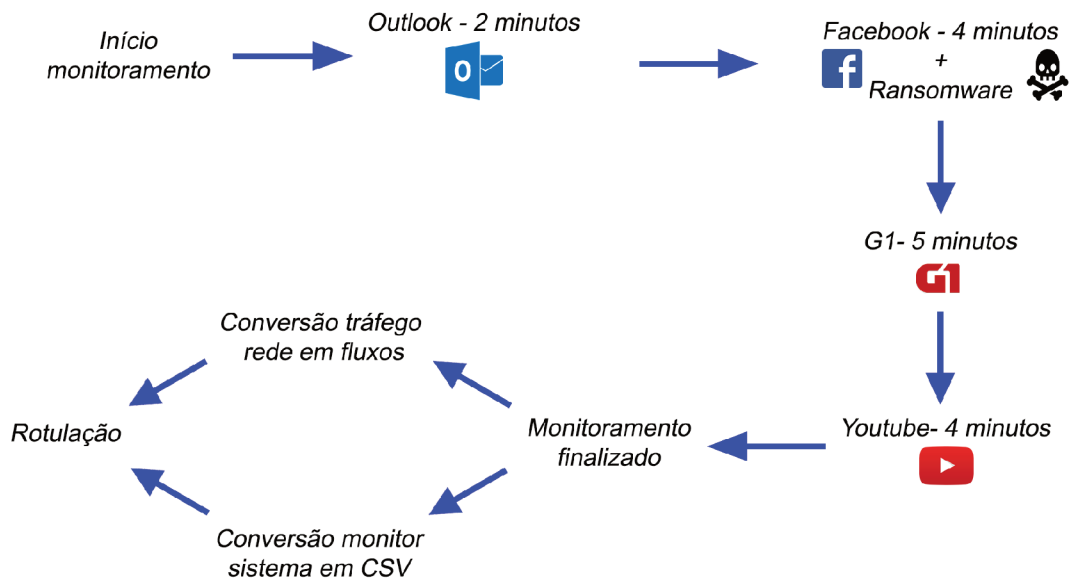


Figura 2 – Passo a passo da execução.

A seguir, as famílias de *ransomware* escolhidas para os experimentos serão descritas. Todos os arquivos executáveis foram encontrados no site <www.malware-traffic-analysis.net>.

- **WannaCry:** em maio de 2016 diversos setores como telecomunicação, saúde, governamental, produção de gás etc. sofreram com ataques massivos do *WannaCry*, cerca de 300.000 sistemas em 150 países foram afetados. Esse *ransomware* utiliza da falha *EternalBlue* e *DoublePulsar* que foram vazados por um grupo chamado *The Shadow Brokers*. *EternalBlue* explora a falha no SMB através da porta 445 do protocolo TCP e é usado para se propagar para outras máquinas e sistemas, já o *DoublePulsar* é um *backdoor*, permitindo o acesso remoto da máquina (AKBANOV et al., 2018).
- **Cerber:** utiliza o algoritmo de criptografia AES e um modelo de negócios *Ransomware as a Service* (RaaS). Sua aparição foi em 2016 e ao infectar a máquina, verifica a localização da mesma através da geolocalização. Caso se encontre em algum desses países: Armênia, Azerbaijão, Bielorrússia, Geórgia, Quirquístão, Cazaquistão, Moldávia, Rússia, Turcomenistão, Tadjiquistão, Ucrânia ou Uzbequistão, o *malware* interrompe seu funcionamento e não continua seu processo de criptografia (KURNIAWAN; RIADI, 2018).
- **GandCrab:** também seguindo o modelo de RaaS, desde sua aparição em 2018, o *malware* tem sido um dos mais ativos entre os *ransomwares*. Isso se deve a grande capacidade dos criadores de atualizarem o código malicioso frequentemente, e portanto, esquivar das detecções (DAHAN, 2019).

- **TeslaCrypt**: inicialmente com o objetivo de atingir usuários de jogos, criptografando dados específicos, evoluiu para criptografar outros arquivos, como PDF, fotos, vídeos etc. Semelhante ao *Cerber*, também utiliza o algoritmo AES-256 para realizar a criptografia e teve ao total três versões lançadas, cada um corrigindo falhas e melhorando o poder de evadir a detecção ([Sophia Wang, 2017](#)).

3.2 Ferramentas utilizadas e desenvolvidas

O ambiente com Windows 7 foi montado sob o software de virtualização VirtualBox. O Wireshark foi o responsável por capturar o tráfego de rede e gerar o arquivo *pcap*², que posteriormente foi convertido para fluxos pelo programa Argus. O Argus, além de uma ferramenta de monitoramento de rede, é também um software de fluxo de rede que tenta solucionar problemas de escalabilidade, *performance*, aplicabilidade, privacidade e utilidade. No programa Argus foi executado o seguinte comando para a conversão do *pcap* em fluxo:

```
argus -r nome_pcap.pcap -w nome_fluxo.argus
```

Assim tem-se um arquivo no formato argus que contém o fluxo que pode ser utilizado para extrair diversas informações através do próprio software, tais como agrupamentos, filtros, expressões regulares etc. Contudo, para realizar a rotulação foi desenvolvido um sistema em Python que recebe como entrada um arquivo *CSV*. É possível converter o fluxo em *CSV* pelo próprio Argus através do comando:

```
ra -r nome_fluxo.argus -A -c ',' > nome_csv.csv
```

Já para o monitoramento do uso do sistema foi utilizado o Monitor de Recursos nativo do próprio Windows. Após obter a saída, os dados da leitura também foram rotulados por um sistema desenvolvido em Python e, portanto, foram convertidos para *CSV* utilizando o seguinte comando:

```
relog monitor.blg -f csv -o monitor_csv.csv
```

A ferramenta desenvolvida em Python realiza a rotulação, adicionando ao final de cada *CSV* na coluna *Type* um valor correspondente ao tipo de tráfego: *Normal* para tráfego de rede não malicioso ou *Malicious* para tráfego de rede malicioso. Já para o monitoramento do computador, o valor no campo *Type* é "1" para leitura normal e "2" para leitura maliciosa.

² *Pcaps* são arquivos com tráfego de rede gerados pelo software Wireshark.

4 Resultados

Em todos os experimentos com as quatro famílias diferentes de *ransomware* foram gerados um arquivo rotulado contendo o tráfego de rede e quatro arquivos rotulados contendo o monitoramento do computador: memória, disco rígido, processador e sistema. O presente capítulo será dividido em seis seções. As quatro primeiras são referentes a apresentação de detalhes dos experimentos para cada um dos *ransomwares*. As duas últimas seções discutem a estratégia usada para inserir os rótulos (tráfego normal/tráfego malicioso) no arquivo CSV e uma aplicação de análise dos dados usando algoritmos de inteligência artificial.

4.1 Experimento 1 - *WannaCry*

A execução do experimento pode ser dividida da seguinte forma:

Início	Fim	Tráfego
15:04	15:06	Outlook.com
15:06	15:10	Facebook.com + execução <i>WannaCry</i>
15:10	15:15	G1.com
15:15	15:19	Youtube.com

Tabela 1 – Tabela com a descrição do tráfego de rede do *WannaCry*

Comparando a quantidade de fluxos normais e maliciosos, obtém-se a seguinte tabela:

Tipo	Quantidade
Normal	3336
Malicioso	22274

Tabela 2 – Tabela comparando quantidade de cada tipo de fluxo do *WannaCry*

4.1.1 Tráfego de rede

É possível notar que a quantidade de fluxos maliciosos é bem maior que os normais. Portanto a atividade de rede do *WannaCry* é bem intensa e pode ser explorada para realizar a detecção.

A rotulação do tráfego do *WannaCry* baseou-se nos seguintes aspectos:

- **Rede TOR:** em seu artigo, ([AKBANOV et al., 2018](#)) cita que o *ransomware* realiza diversas comunicações com IP's da rede TOR através das portas 9001 e 9050.

Portanto, no fluxo gerado foram selecionados todos os IP's que em sua comunicação utilizavam essas portas e verificados se eles estavam presentes na lista de *nodes* da rede TOR através do site <https://www.dan.me.uk>. Foram encontrados cinco IP's, dentre eles: 5.79.79.133, 87.120.36.210, 154.35.175.225, 171.25.193.9, 193.23.244.244. Na rotulação, esses IP's foram considerados maliciosos juntamente com qualquer comunicação nas portas 9001, 9050, 9030 e 9101.

- Todo tráfego na porta 445 (Microsoft-ds), 137, 138 e 139 foi considerado malicioso, uma vez que o *Server Message Block* (SMB) utiliza essas portas e a principal característica do *WannaCry* é utilizar a vulnerabilidade do SMB para se propagar.
- Tráfego do protocolo ARP também foi considerado malicioso já que o *WannaCry* gerou uma quantidade excessiva de requisições desse tipo e logo pode ser usado como um indício de infecção.

4.1.2 Monitoramento do computador

Para realizar a rotulação do monitoramento do computador durante a execução do *WannaCry*, foi estabelecido malicioso qualquer resultado entre 15:06:00 e 15:12:15. As Figuras 3, 4, 5 e 6 mostram os resultados do monitoramento da memória, disco rígido, processador e sistema respectivamente. Nota-se que praticamente todos os contadores do monitoramento do computador, com exceção do processador, tiveram alguma oscilação ao iniciar a execução do *WannaCry*.

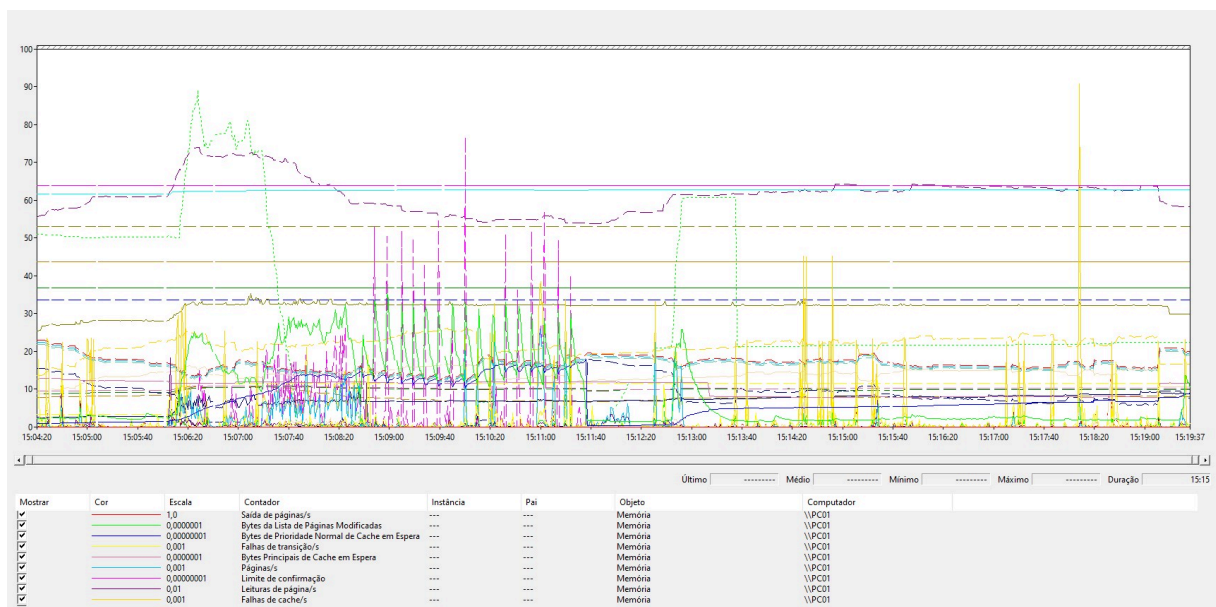


Figura 3 – Gráfico do monitoramento da memória.

No monitoramento da memória há um forte aumento na quantidade de memória, em bytes, atribuída à lista de páginas modificadas, representado pela linha verde contínua. Outro contador que também se destacou foi a linha azul claro contínua, representando uma alta na quantidade de páginas lidas do ou gravadas no disco para resolver falhas de páginas física. Observa-se que as falhas de cache, identificado pela linha amarela contínua, também oscilaram com a execução do *malware*, onde páginas requisitadas não são encontradas e devem ser buscadas ou do disco ou de outra parte da memória.

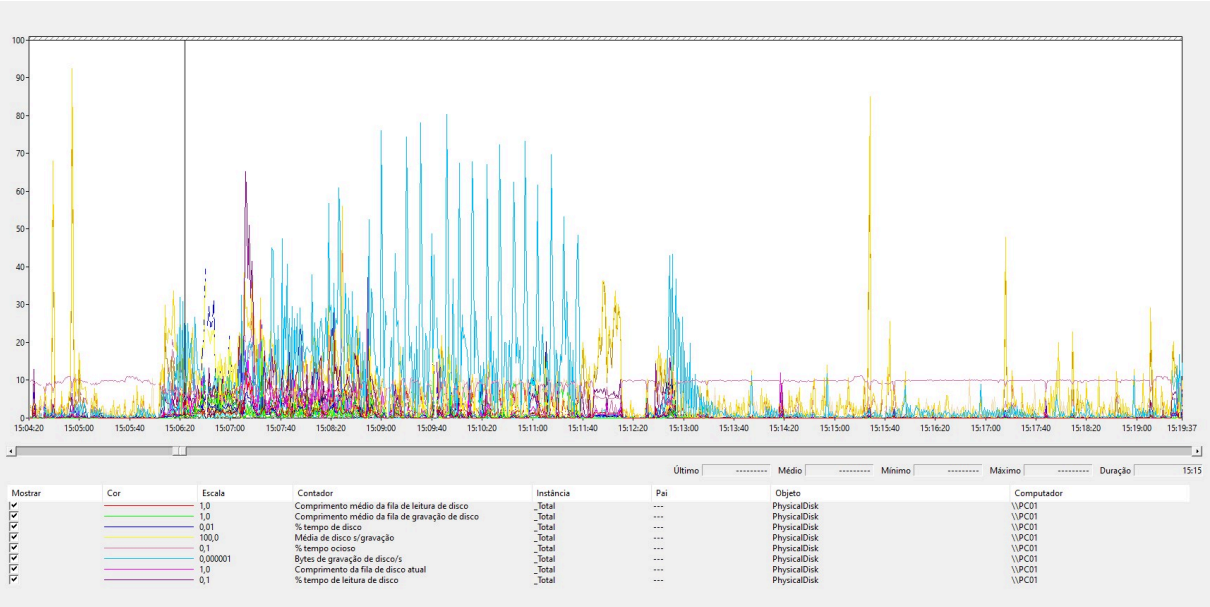


Figura 4 – Gráfico do monitoramento do disco rígido.

Já no monitoramento do disco rígido, percebe-se uma oscilação na linha amarela contínua que representa o tempo médio, em segundos, das gravações de dados para o disco. Outro aumento foi a taxa na qual bytes são transferidos para o disco durante as operações de gravação, identificado pela linha azul contínua. A linha roxa também se destacou e representa a porcentagem de tempo em que uma unidade de disco estava ocupada atendendo solicitações de leitura.

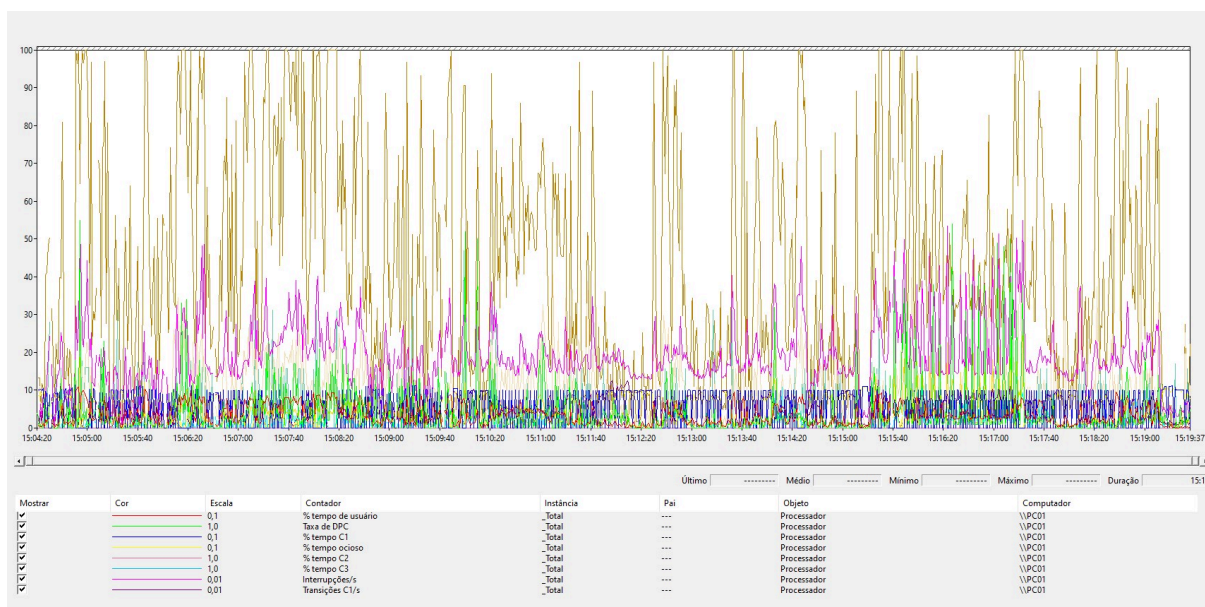


Figura 5 – Gráfico do monitoramento do processador.

Para o monitoramento do processador não foi possível notar nenhum comportamento anormal ao executar o *ransomware*.

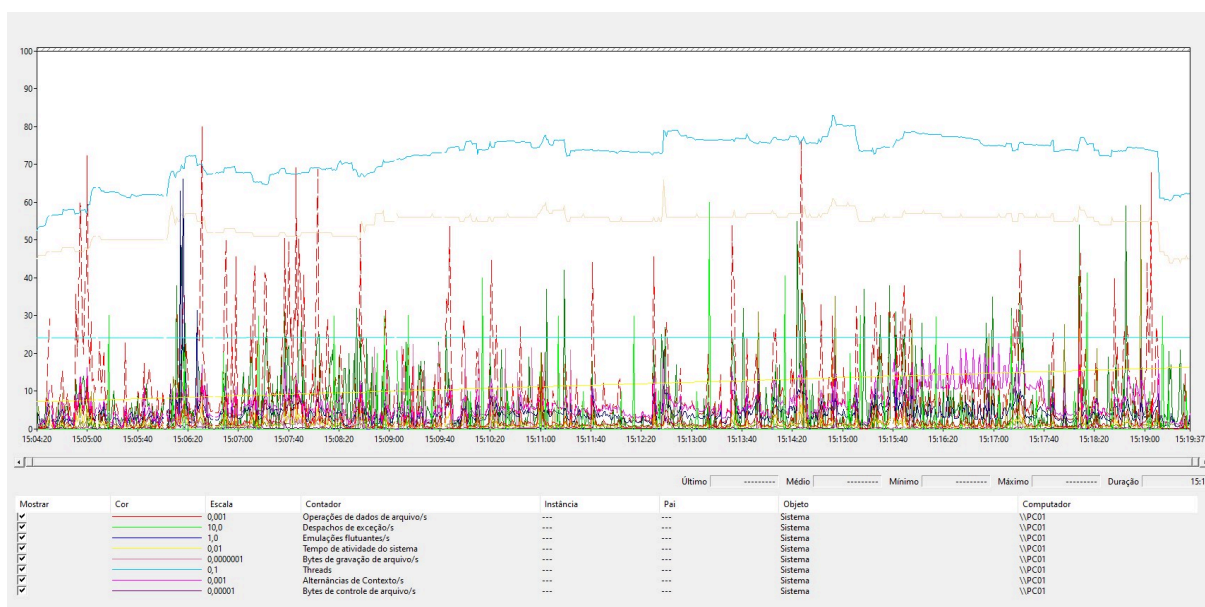


Figura 6 – Gráfico do monitoramento do sistema.

No monitoramento do sistema, observa-se um aumento na taxa de bytes que são escritos e lidos devido a solicitações do sistema de arquivos e também um aumento na frequência de chamadas de sistema, representados pelas linhas rosa claro contínua, azul escuro contínua e marrom contínua respectivamente.

4.2 Experimento 2 - *TeslaCrypt*

A execução do experimento pode ser dividida da seguinte forma:

Início	Fim	Tráfego
22:19	22:21	Outlook.com
22:21	22:25	Facebook.com + execução <i>TeslaCrypt</i>
22:25	22:30	G1.com
22:30	22:34	Youtube.com

Tabela 3 – Tabela com a descrição do tráfego de rede do *TeslaCrypt*

Comparando a quantidade de fluxos normais e maliciosos, obtém-se a seguinte tabela:

Tipo	Quantidade
Normal	2608
Malicioso	18

Tabela 4 – Tabela comparando quantidade de cada tipo de fluxo do *TeslaCrypt*

4.2.1 Tráfego de rede

Para o *TeslaCrypt*, o tráfego de rede não é muito intenso, contendo requisições maliciosas para somente dois endereços IP's: 92.242.140.6 e 107.163.164.171, pertencentes aos domínios *13343225565.com* e *aforexvn.com* respectivamente. Tais IP's foram obtidos ao analisar o tráfego somente do *ransomware*, sem nenhum outro tráfego paralelo.

4.2.2 Monitoramento do computador

Para realizar a rotulação do monitoramento de sistema do *TeslaCrypt*, foi estabelecido malicioso qualquer resultado entre 22:21:00 e 22:27:30. As Figuras 7, 8, 9 e 10 mostram os resultados do monitoramento da memória, disco rígido, processador e sistema respectivamente.

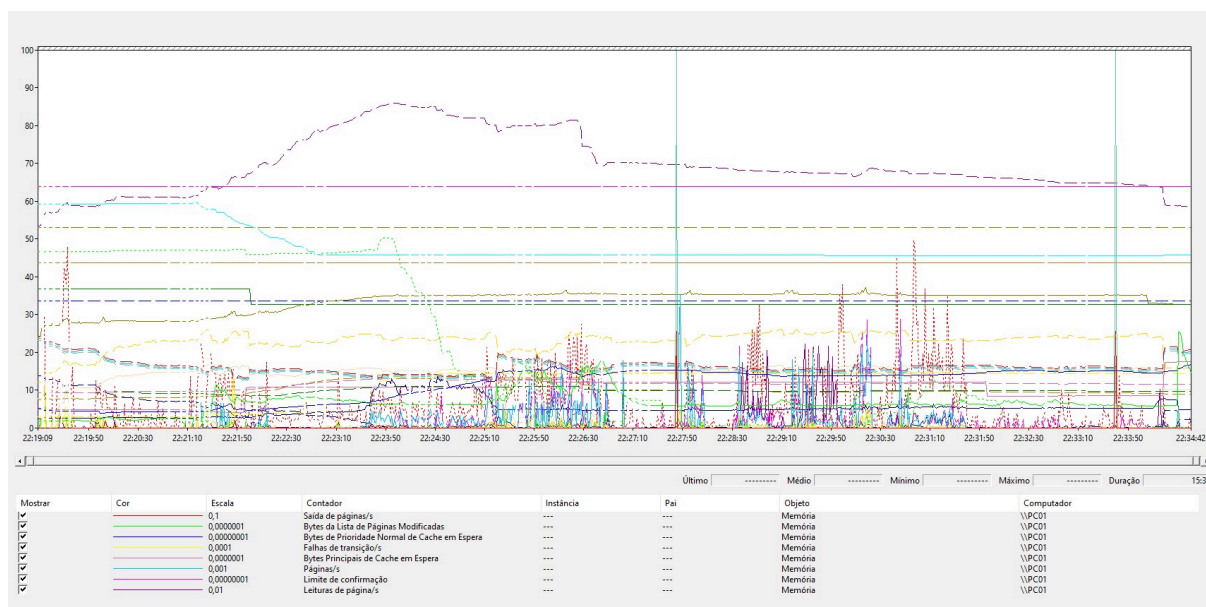


Figura 7 – Gráfico do monitoramento da memória.

Para o monitoramento da memória durante a execução do *TeslaCrypt* observa-se algumas oscilações em contadores. A primeira é a quantidade de bytes atribuída à lista de páginas modificadas, representado pela linha verde contínua. A segunda é o tamanho da parte do cache do arquivo de sistema que está ativa na memória, representado pela linha azul escuro contínua. Por fim temos a oscilação do número de chamadas para alocar espaço na área da memória para objetos que não podem ser gravados em disco no momento, identificado pela linha roxa pontilhada.

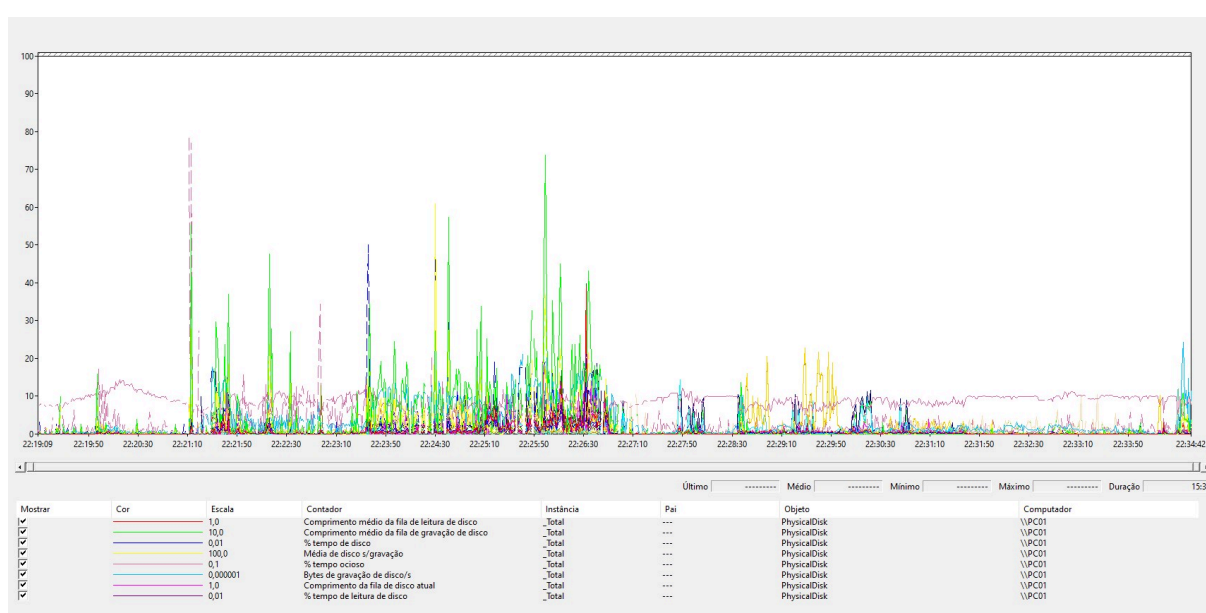


Figura 8 – Gráfico do monitoramento do disco rígido.

No monitoramento do disco rígido, aumento expressivo no comprimento médio da fila de gravação de disco, ou seja, no número de solicitações de gravação enfileiradas para o disco, representado pela linha verde contínua. Outro contador com oscilação foi a taxa na qual bytes são transferidos para o disco, representado pela linha azul claro contínua.

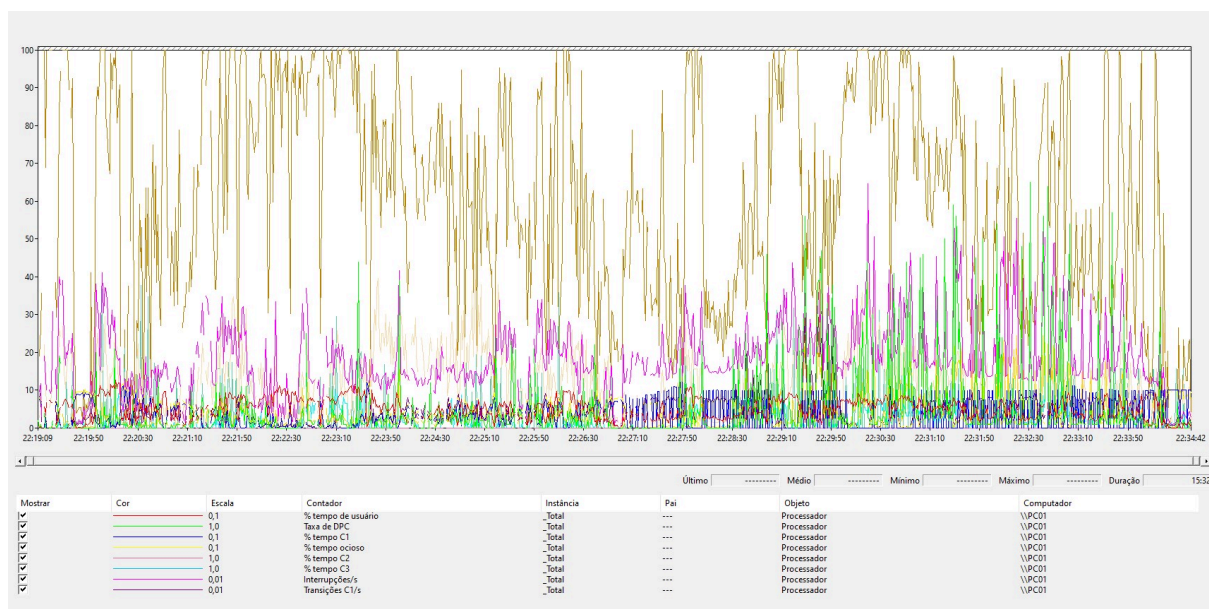


Figura 9 – Gráfico do monitoramento do processador.

Semelhante ao *WannaCry*, não foi possível identificar nenhum comportamento anormal após a execução do *ransomware*.

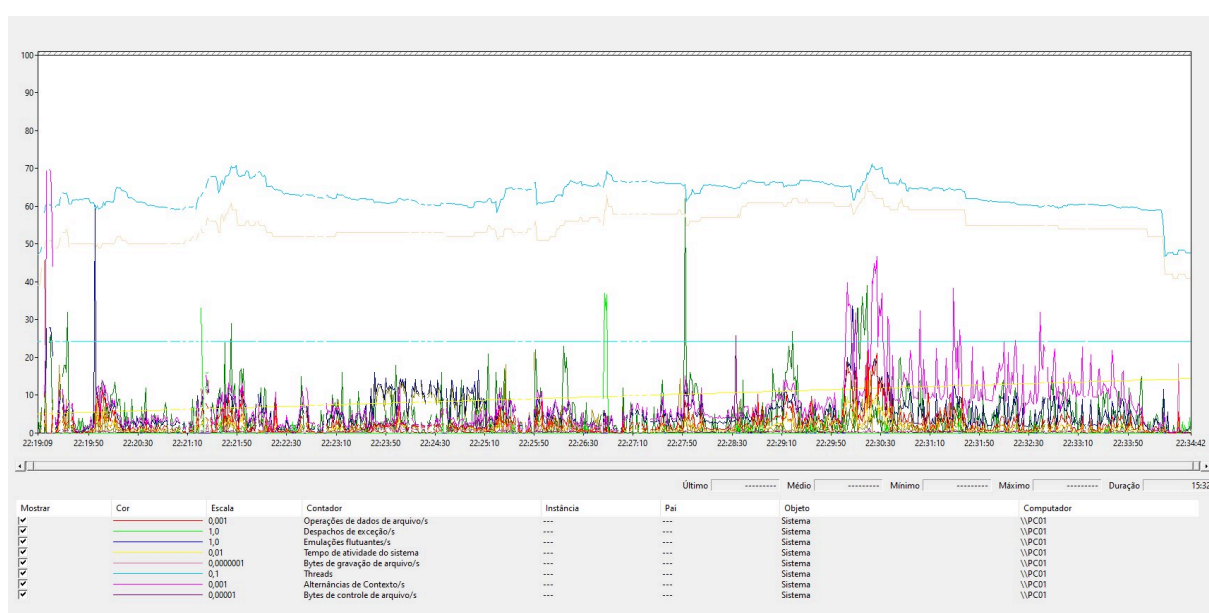


Figura 10 – Gráfico do monitoramento do sistema.

Dois contadores referentes ao monitoramento do sistema do *TeslaCrypt* que oscilaram no período foram a taxa em que bytes são gravados para atender solicitações do sistema de arquivos e a taxa combinada de operações do sistema de arquivos que não são de leituras ou gravações. Esses dois contadores são representados pelas linhas rosa claro contínua e marrom escuro contínua.

4.3 Experimento 3 - *GandCrab*

A execução do experimento pode ser dividida da seguinte forma:

Início	Fim	Tráfego
12:56	12:58	Outlook.com
12:58	13:02	Facebook.com + execução <i>GandCrab</i>
13:02	13:07	G1.com
13:07	13:11	Youtube.com

Tabela 5 – Tabela com a descrição do tráfego de rede do *GandCrab*

Comparando a quantidade de fluxos normais e maliciosos, obtém-se a seguinte tabela:

Tipo	Quantidade
Normal	2444
Malicioso	353

Tabela 6 – Tabela comparando a quantidade de cada tipo de fluxo do *GandCrab*

4.3.1 Tráfego de rede

O tráfego pós infecção do *GandCrab* possui uma característica diferente dos demais: apresenta diversas requisições a servidores aparentemente legítimos. Diversos sites na captura são requisitados, muitos deles pertencentes a hotéis e lojas, um exemplo é o: *www.arbezie-hotel.com*. Uma suposição é que o *ransomware* gera tráfego legítimo para evadir possíveis sistemas de detecção. Na rotulação, todas essas requisições foram consideradas como indícios de infecção e marcadas como maliciosas.

4.3.2 Monitoramento do computador

Para realizar a rotulação do monitoramento de sistema do *GandCrab*, foi estabelecido malicioso qualquer resultado entre os intervalos 12:58:10 até 13:00:25 e 13:04:00 até 13:06:40. As Figuras 11, 12, 13 e 14 mostram os resultados do monitoramento da memória, disco rígido, processador e sistema respectivamente.

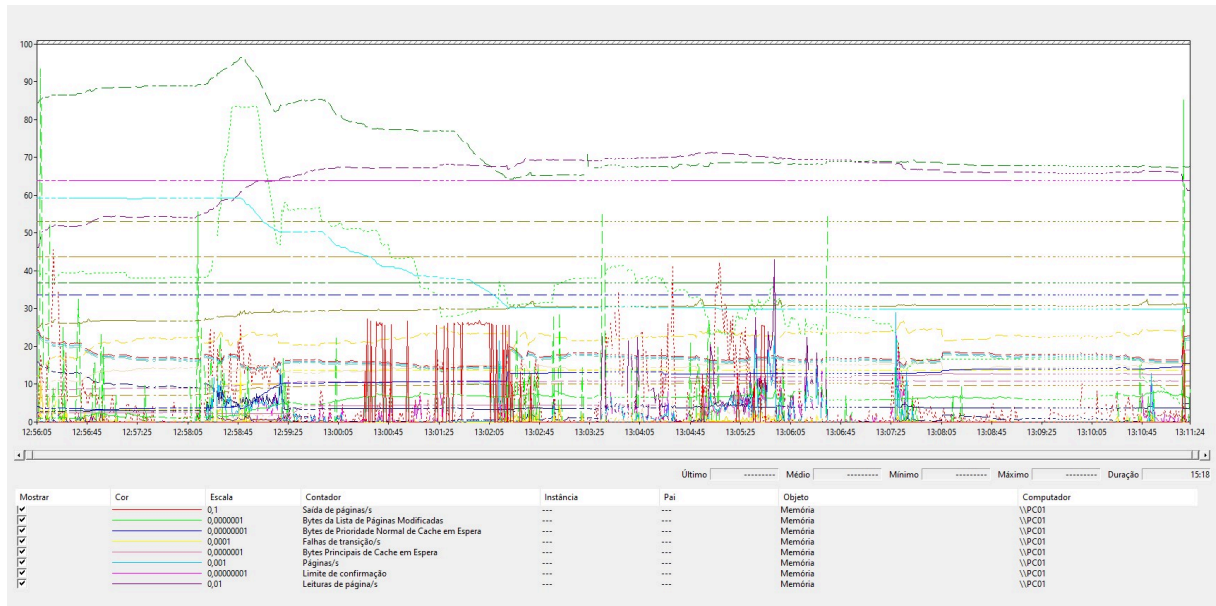


Figura 11 – Gráfico do monitoramento da memória.

Para o monitoramento de memória do *GandCrab* um contador que teve uma oscilação expressiva foi a taxa em que as páginas são lidas do ou gravadas no disco afim de resolver falhas de páginas. Tal contador é representado pela linha azul claro contínuo.

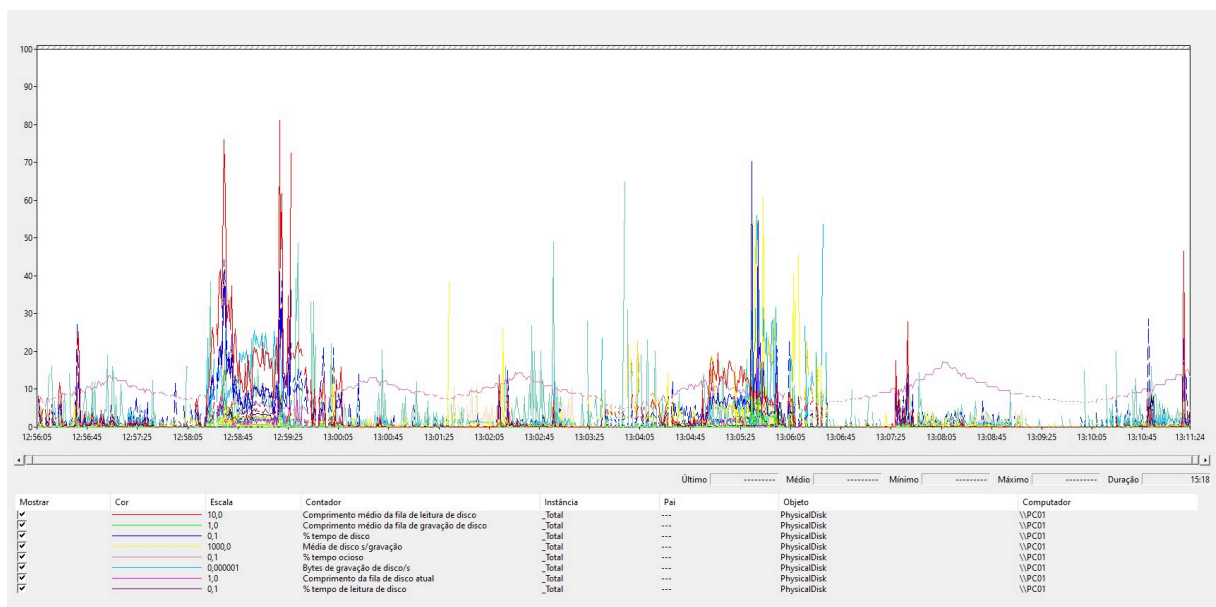


Figura 12 – Gráfico do monitoramento do disco rígido.

A partir de alguns contadores é possível notar que o *GandCrab* teve dois picos de criptografia dos dados em intervalos diferentes. Tais contadores que indicam esse fato são:

- A taxa à qual bytes são transferidos para o disco, representado pela linha azul contínua.
- A porcentagem de tempo na qual a unidade de disco estava ocupada com requisições de leitura ou gravação, identificado pela linha azul escuro contínua.

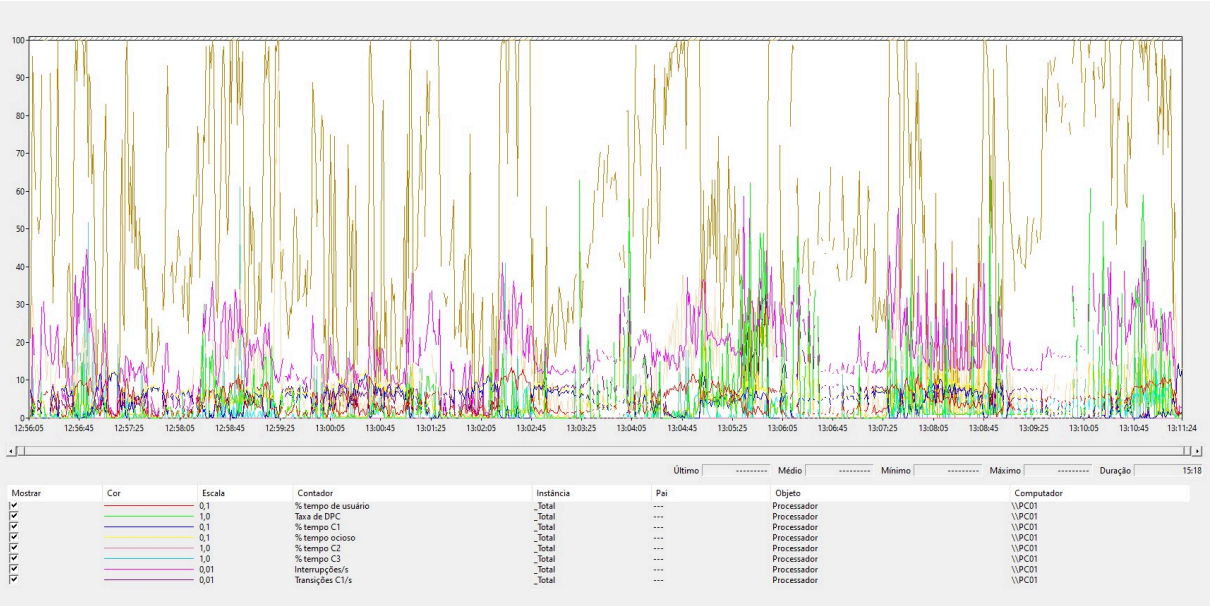


Figura 13 – Gráfico do monitoramento do processador.

Semelhante com os dois *ransomwares* acima, não foi identificado nenhum contador com oscilações que indicassem atividade suspeita.

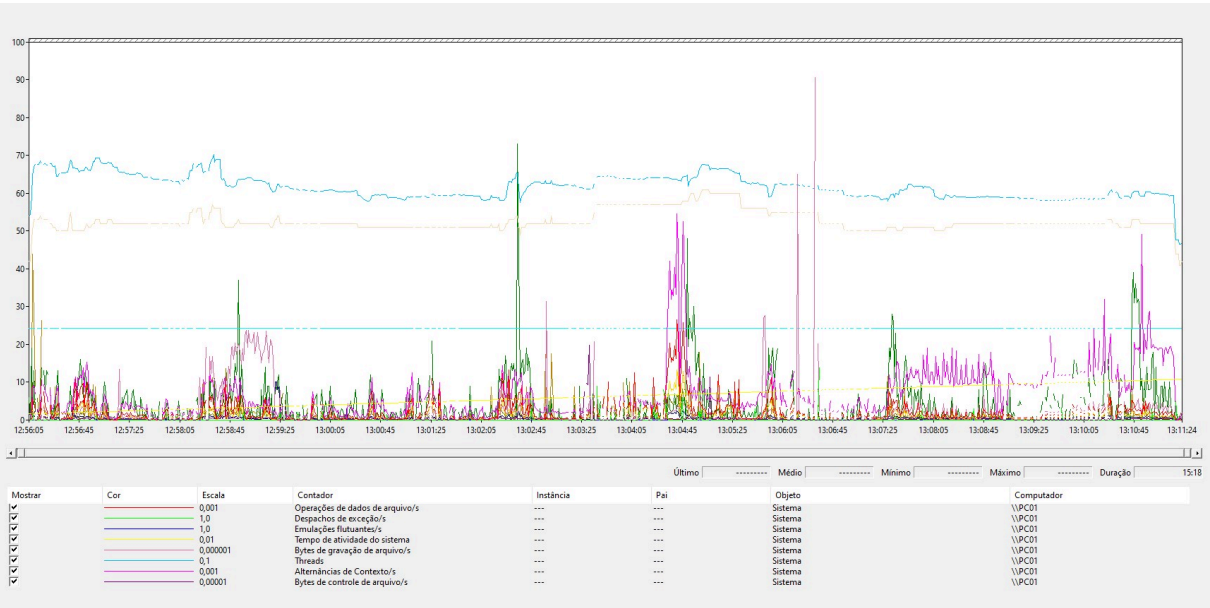


Figura 14 – Gráfico do monitoramento do sistema.

Nesse monitoramento, dois contadores representados pelas linhas rosa clara contínua e marrom contínua obtiveram uma oscilação. Eles são referentes a taxa em que bytes são gravados ou lidos para satisfazer requisições do sistema de arquivos.

4.4 Experimento 4 - *Cerber*

A execução do experimento pode ser dividida da seguinte forma:

Início	Fim	Tráfego
22:15	22:17	Outlook.com
22:17	22:21	Facebook.com + execução <i>Cerber</i>
22:21	22:16	G1.com
22:26	22:30	Youtube.com

Tabela 7 – Tabela com a descrição do tráfego de rede do *Cerber*

Comparando a quantidade de fluxos normais e maliciosos, obtém-se a seguinte tabela:

Tipo	Quantidade
Normal	3767
Malicioso	2196

Tabela 8 – Tabela comparando a quantidade de cada tipo de fluxo do *Cerber*

4.4.1 Tráfego de rede

O *Cerber*, semelhante ao *WannaCry*, possui uma grande quantidade de fluxos maliciosos se comparado com o *GandCrab* e *TeslaCrypt*. Essa família realiza requisições para os seguintes domínios: *btc.blockr.io*, *api.blockcypher.com*, *chain.so* e *bitaps.com* (YAMBAO, 2016). Além disso, também conecta a diversos servidores através do protocolo UDP na porta 6893 (PRODUCTS, 2017).

4.4.2 Monitoramento do computador

Para realizar a rotulação do monitoramento de sistema do *Cerber*, foi estabelecido malicioso qualquer resultado entre 22:17:00 e 22:21:00. As Figuras 15, 16, 17 e 18 mostram os resultados do monitoramento da memória, disco rígido, processador e sistema respectivamente.

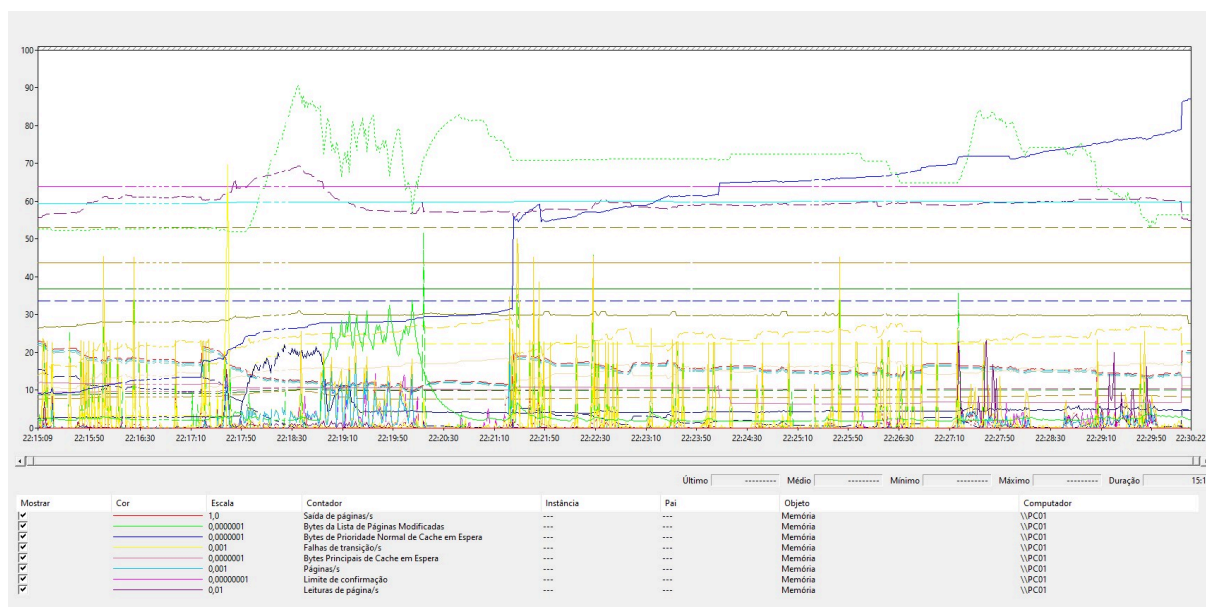


Figura 15 – Gráfico do monitoramento da memória.

Semelhante ao *WannaCry* e ao *GandCrab*, o contador referente a taxa em que páginas são lidas do ou gravadas no disco para resolver falhas de página física também sofreu uma forte oscilação ao executar o *Cerber*. Ele é representado pela linha azul claro contínua no gráfico acima. Outro ponto observado foi o aumento do tamanho da parte do cache do arquivo de sistema representado pela linha azul escuro contínua.

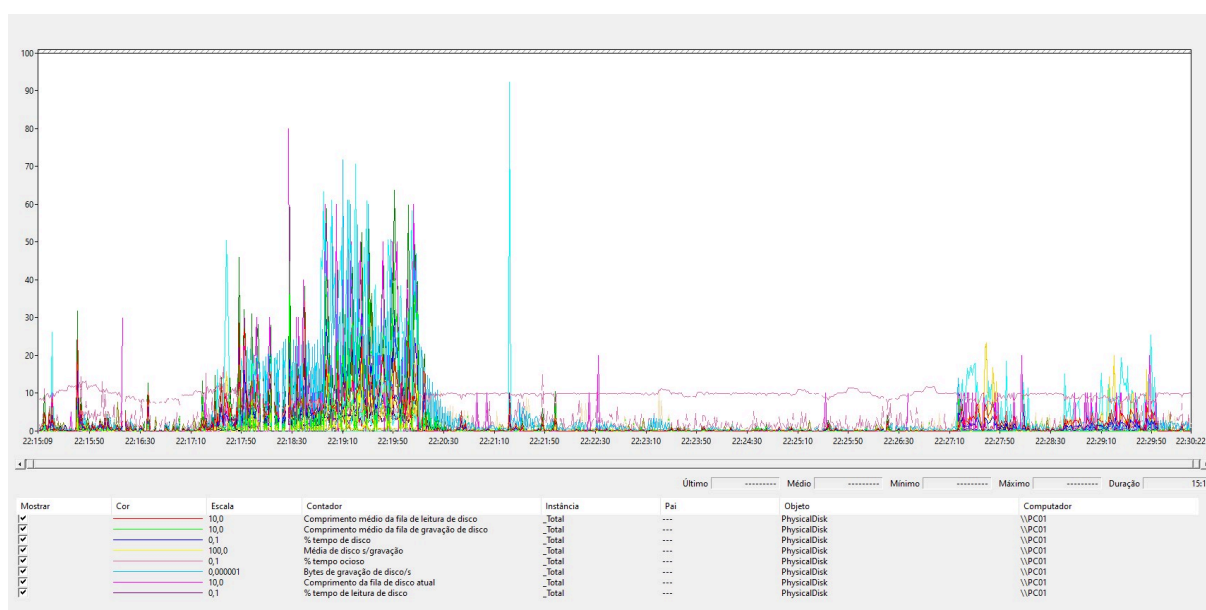


Figura 16 – Gráfico do monitoramento do disco rígido.

Praticamente quase todos os contadores sofreram uma alta variação ao executar

o *Cerber*. Um deles é a linha verde contínua, representando o comprimento médio da fila de gravação do disco.

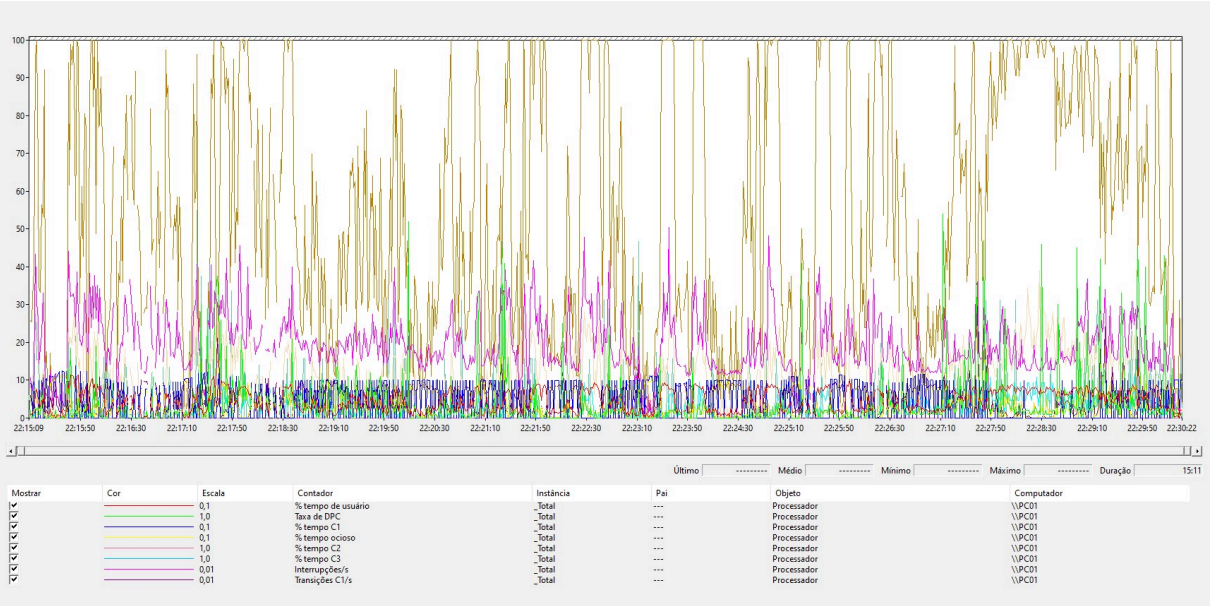


Figura 17 – Gráfico do monitoramento do processador.

Não foi possível identificar nenhuma oscilação anormal.

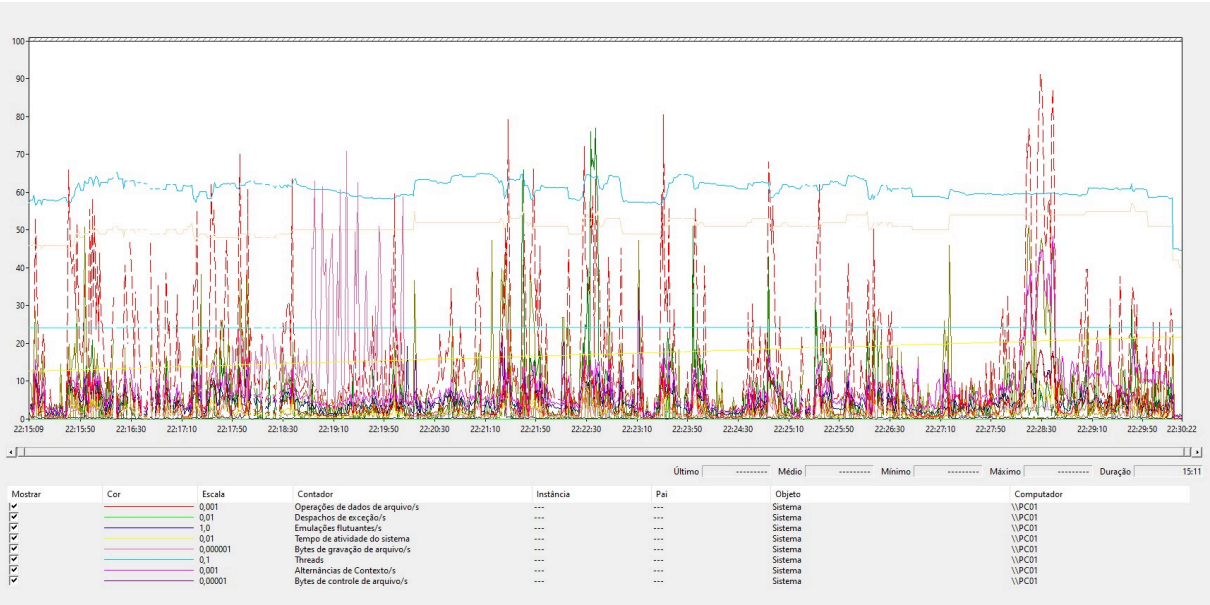


Figura 18 – Gráfico do monitoramento do sistema.

Um contador que se destacou dos outros foi a taxa de bytes que são gravados devido à solicitações do sistema de arquivos, representado pela linha rosa contínua.

4.5 Arquivos rotulados

Para a rotulação dos dados referente ao monitoramento do computador, em todos os casos foi escolhido uma janela de tempo onde qualquer resultado é considerado malicioso. Tal janela possui a execução do *malware* como horário de início e quando já não havia mais arquivos para serem criptografados como horário final. O campo adicionado no *CSV* para indicar o tipo de cada leitura do monitoramento chama-se *Type* e possui o valor "1" para uma leitura normal e "2" para maliciosa. Foi preciso utilizar valores inteiros ao invés de palavras como "*malicious*" e "*normal*" como nos fluxos de rede, uma vez que para aplicar os algoritmos de aprendizado de máquina descritos no próximo tópico tais valores precisavam ser do tipo inteiro.

A seguir um exemplo de um arquivo rotulado (Figura 19) para o tráfego de rede e outro para o monitor de desempenho (Figura 20).

```
StartTime,Flgs,Proto,SrcAddr,Sport,Dir,DstAddr,Dport,SrcPkts,DstPkts,SrcBytes,DstBytes,Type
15:04:25.965193,e,udp,10.0.2.21,55405,<->,8.8.8.8,domain,1.0,1.0,74.0,90.0,NORMAL
15:04:25.997205,e,udp,10.0.2.21,55406,<->,172.217.162.100,443,33.0,35.0,5814.0,5856.0,NORMAL
15:04:27.991603,e,udp,10.0.2.21,56725,<->,8.8.8.8,domain,1.0,1.0,71.0,87.0,NORMAL
```

Figura 19 – Parte de um arquivo de fluxo rotulado.

Os atributos listados no cabeçalho da Figura 19 acima são definidos por padrão pelo Argus. Contudo, é possível modificar as configurações do software para gerar fluxos com diferentes atributos, tais como *smac* e *dmac*, representando o endereço MAC de origem e destino respectivamente. A tabela abaixo descreve os atributos padrões do Argus.

StartTime	Início do fluxo
Flgs	Flags observados no fluxo
Proto	Protocolo do fluxo
SrcAddr	Endereço IP de origem
Sport	Porta de origem
Dir	Direção do fluxo
DstAddr	Endereço IP de destino
Dport	Porta de destino
SrcPkts	Quantidade de pacotes provenientes da origem
DstPkts	Quantidade de pacotes provenientes do destino
SrcBytes	Quantidade de bytes provenientes da origem
DstBytes	Quantidade de bytes provenientes do destino

Tabela 9 – Tabela com as descrições dos atributos gerados pelo Argus

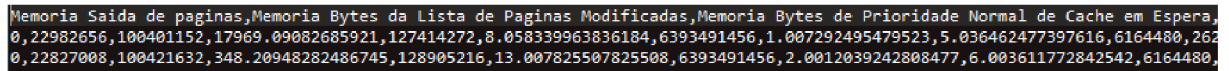


Figura 20 – Parte de um arquivo de desempenho do computador rotulado.

A Figura 20 está representando apenas uma pequena parte do arquivo rotulado para a leitura do monitoramento disco rígido, uma vez que se abrangeu uma alta quantidade de métricas.

A Tabela 10 representa a quantidade de métricas utilizadas em cada um dos quatro tipos de monitoramento do computador.

Tipo	Quantidade
Memória	35
Disco Rígido	22
Processador	16
Sistema	18

Tabela 10 – Tabela com a quantidade de métricas em cada tipo de monitoramento do computador

Exemplos das métricas citadas acima são:

- **Memória**

- **Bytes de cache:** é o tamanho, em bytes, da parte do cache do arquivo de sistema que está ativa na memória.
- **Páginas:** é a taxa em que as páginas são lidas do ou gravadas no disco para resolver as falhas de páginas física.

- **Disco rígido**

- **Bytes de gravação de disco:** é a taxa em que os bytes são transferidos para o disco durante operações de gravação.
- **Comprimento médio da fila de gravação de disco:** é o número médio e solicitações de gravação enfileiradas para o disco.

- **Processador**

- **Interrupções:** é a taxa média, em incidentes por segundo, em que o processador recebeu e atendeu a interrupções de hardware.
- **Porcentagem de tempo ocioso:** é a porcentagem de tempo em que o processador está ocioso.

- **Sistema**

- **Chamadas do sistema:** é a frequência de chamadas de rotinas de serviço do sistema operacional por parte de todos os processos em execução.
- **Processos:** é o número de processos no computador.

4.6 Classificação do tráfego usando aprendizado de máquina - Exemplo de aplicação do trabalho

Um ponto positivo de obter arquivos no formato *CSV* é a capacidade de trabalhar com esses dados em diversas ferramentas e tecnologias. Um delas é o aprendizado de máquina, que na visão de (ALPAYDIN, 2010) é responsável por programar computadores para otimizar um critério de desempenho usando dados de exemplos ou experiências anteriores. Em seu livro, (ALPAYDIN, 2010) comenta sobre a capacidade de reunir milhões de exemplos de e-mails e poder dizer se um determinado e-mail é *spam* ou não através de reconhecimento de padrões. Ou seja, pode-se utilizar o aprendizado de máquina para realizar precisões assertivas. Outro caso também citado é a capacidade de se obter informações valiosas que são extraídas de uma grande base de dados, como por exemplo, qual a probabilidade de um cliente consumir um produto X dado que comprou o produto Y. Em uma comparação do aprendizado de máquina com programação direta, tem-se que a primeira é orientado a dados e consegue analisar muitas informações de uma vez, enquanto que a segunda além de não conseguir processar uma grande quantidade de informações, também sofre interferências de fatores humanos (SCHAPIRE, 2008).

Foram aplicados dois algoritmos de aprendizado de máquina: *Random Forest* e KNN no resultado do monitoramento do disco rígido do *WannaCry*.

Random Forest é um algoritmo que utiliza da combinação de vários classificadores e da união de seus resultados para produzir uma saída, ou seja, é um algoritmo *ensemble* (LOPES et al., 2017). Algoritmos *ensemble* tendem a produzir resultados mais confiáveis do que algoritmos que utilizam de apenas um classificador. O *Random Forest* utiliza da geração de várias árvores de decisão pequenas e simples para a geração de uma árvore única e maior (LOPES et al., 2017).

O KNN (*K Nearest Neighbors*) é algoritmo que utiliza do conceito de similaridade para realizar classificação e regressão. Através de uma função de similaridade, que geralmente é a distância euclidiana, é possível classificar uma nova entrada baseado em sua semelhança com dados já aprendidos. Ou seja, a estratégia é encontrar os k-vizinhos mais próximos dessa nova entrada e atribuir a ela a mesma classes desses vizinhos.

O objetivo desse breve estudo de caso é utilizar um modelo de classificação supervisionado para dizer se, a partir de dados de monitoramento do disco rígido, uma

máquina está ou não infectada com o *WannaCry*. O modelo é supervisionado, uma vez que tem-se os resultados do monitoramento devidamente rotulados em leituras normais ou maliciosas. Ao aplicar os algoritmos *Random Forest* e KNN foi utilizado como base de treinamento cerca de 80% da massa de dados e os testes foram realizados com 20% dos dados.

Nota-se que a aplicação do *Random Forest* para identificar um tráfego malicioso usando como base dados de monitoramento do disco rígido, obteve-se uma taxa de acurácia de 93% conforme (Figura 21). Acurácia é a porcentagem de acerto do algoritmo de forma geral, sobre o todo (SANTANA, 2017), ou seja, dos 20% da massa de dados usados para testes, 93% foram classificados em normal ou malicioso de forma correta. Contudo, um lado negativo é a grande quantidade de nós na árvore gerada (Figura 23), deixando-a custosa. Por exemplo, no pior dos casos seria necessário percorrer 11 nós até chegar em uma conclusão sobre a classificação da leitura. Para a construção dessa árvore, foram extraídas as variáveis de maior relevância, conforme Figura 22.

	precision	recall	f1-score	support
1	0.93	0.94	0.94	105
2	0.92	0.91	0.92	79
accuracy			0.93	184
macro avg	0.93	0.93	0.93	184
weighted avg	0.93	0.93	0.93	184

Figura 21 – Resultado da fase de teste do algoritmo Random Forest.

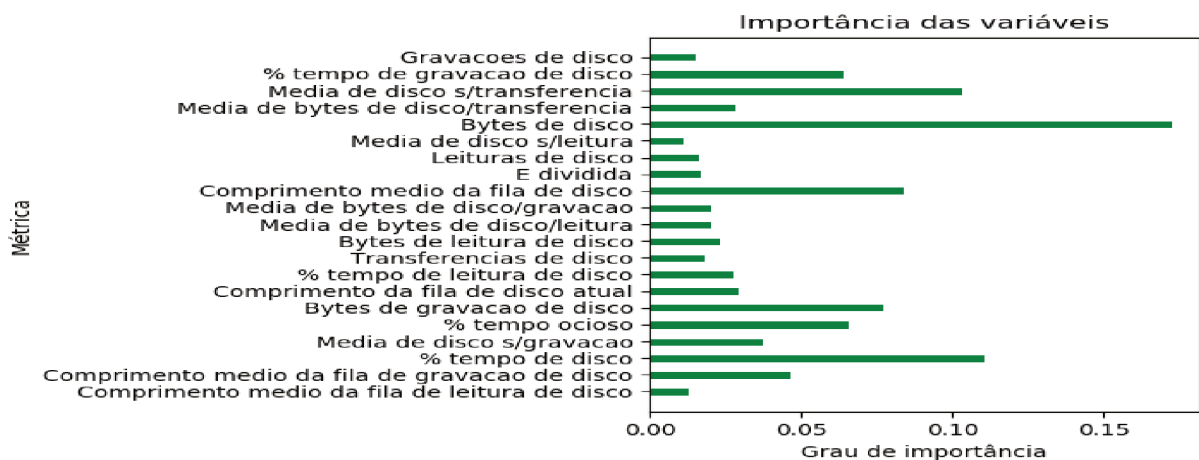


Figura 22 – Variáveis com maior importância na geração da árvore.

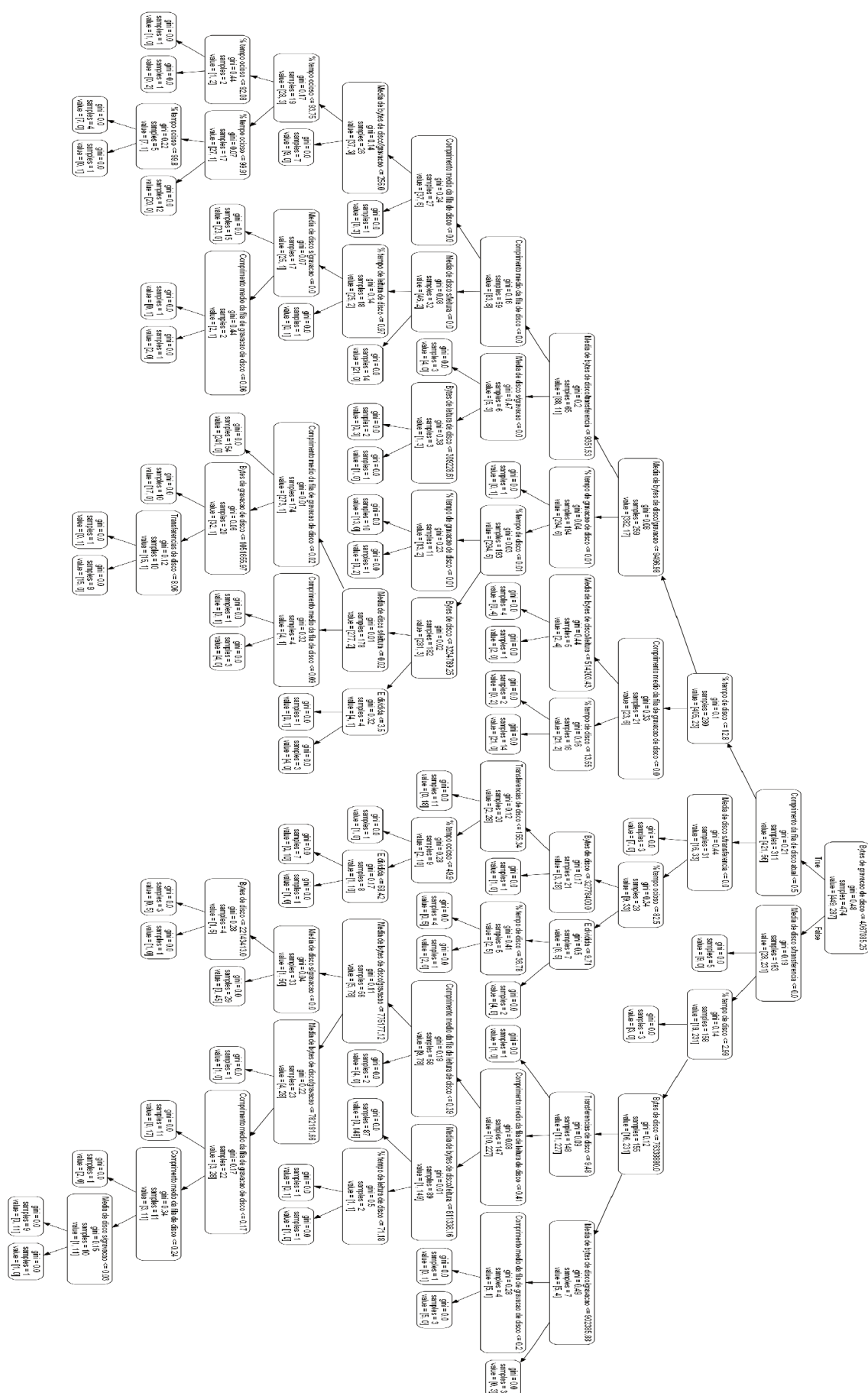


Figura 23 – Árvore de decisão gerada usando *Random Forest*

A variável "Bytes de disco", que representa a taxa à qual bytes são transferidos para ou do disco, foi a que mais se destacou, com uma importância de 17%. Em segundo lugar tem-se a porcentagem de tempo em que o disco estava ocupado atendendo solicitações de leitura ou gravação, com uma importância de 11%.

Os resultados da aplicação do algoritmo KNN foram semelhantes, obtendo cerca de 91% de acurácia, ou seja, dos 20% dos dados usados como teste, 91% foram classificados de forma correta. Tal resultado está representado pelo campo "*accuracy*" na Figura 24.

	precision	recall	f1-score	support
1	0.91	0.94	0.92	110
2	0.90	0.86	0.88	74
accuracy			0.91	184
macro avg	0.91	0.90	0.90	184
weighted avg	0.91	0.91	0.91	184

Figura 24 – Resultado da fase de teste do algoritmo KNN.

Como esse tópico se trata de apenas um breve estudo de caso da aplicação de algoritmos de aprendizado de máquina, é importante aprofundar os estudos em como ajustar os parâmetros dos algoritmos para maior performance, tanto do *Random Forest* quanto do KNN. Parâmetros tais como a quantidade de árvores utilizadas no *Random Forest* e qual o valor de K mais apropriado para o KNN. Além disso, diferentes tempos de monitoramento também podem ser utilizados para verificar a oscilação dos resultados.

Os conjuntos de dados neste trabalho estão prontos para serem usados em diferentes tipos de abordagens experimentais, inclusive em outros algoritmos, tais como SVM (*Support Vector Machine*) e *Naive Bayes* por exemplo.

5 Conclusão

O objetivo desse trabalho foi a criação de conjuntos de dados a partir de quatro diferentes famílias de *ransomware* para que sejam utilizados em pesquisas de outros autores e também na criação e avaliação de ferramentas de segurança da informação, mais especificamente em ferramentas de detecção de *ransomware*.

Durante a criação dos conjuntos, pode-se notar que cada família de *ransomware* possui um tráfego de rede distinto principalmente baseado na sua capacidade de se propagar e evadir. Enquanto o *WannaCry* gera diversas requisições TCP na porta 445 o *Cerber* realiza requisições UDP na porta 6893. Já o *GandCrab* aparentemente usa tráfego normal para tentar evadir possíveis detecções, enquanto que o *TeslaCrypt* quase não gerou nenhum tráfego.

Em relação ao monitoramento do computador, nota-se que todas as famílias obtiveram resultados semelhantes, com alto uso do disco e memória nos primeiros minutos após a execução do *ransomware*. Inclusive nota-se que o monitoramento do processador indicou que não houve oscilações consideráveis no uso do processador ao executar os *ransomwares*. Ao aplicar algoritmos de aprendizado de máquina para a criação de modelos de classificação, foi possível extrair diversas informações importantes sobre o problema e que podem ser usadas como indícios de infecção, por exemplo.

Como trabalhos futuros, sugere-se a construção de conjuntos de dados para novas famílias de *ransomware* e assim disponibilizar maior base de dados para avaliação de ferramentas de detecção. Outro ponto é a aplicação massiva de outros algoritmos de aprendizado de máquina, tais como SVM e *Naive Bayes* para obter novos modelos de classificação tanto para os fluxos de rede quanto para o monitoramento do computador e comparar quais os melhores algoritmos para cada caracterizar cada família de *ransomware*.

Referências

- AKBANOV, M. et al. Static and Dynamic Analysis of WannaCry Ransomware. 2018. Citado 4 vezes nas páginas 12, 16, 18 e 20.
- ALPAYDIN, E. *Introduction to Machine Learning*. 2nd. ed. [S.l.]: The MIT Press, 2010. ISBN 026201243X, 9780262012430. Citado na página 35.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 17799: Tecnologia da informação — técnicas de segurança — código de prática para a gestão da segurança da informação*. Rio de Janeiro, 2005. 132 p. Citado na página 10.
- BULGURCU, B.; CAVUSOGLU, H.; BENBASAT, I. Information security policy compliance: An empirical study of rationality-based beliefs and information security awareness. *MIS Quarterly*, v. 34, p. 523–548, 09 2010. Citado 2 vezes nas páginas 7 e 11.
- CAVUSOGLU, H. et al. Economics of it security management: Four improvements to current security practices. *Communications of the Association for Information Systems*, v. 14, p. 65–75, 01 2004. Citado na página 11.
- CUSACK, G.; MICHEL, O.; KELLER, E. Machine Learning-Based Detection of Ransomware Using SDN. *Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization - SDN-NFV Sec'18*, p. 1–6, 2018. Disponível em: <<http://dl.acm.org/citation.cfm?doid=3180465.3180467>>. Citado na página 7.
- DAHAN, R. O. A. *GANDCRAB'S NEW EVASIVE INFECTION CHAIN*. 2019. <<https://www.cybereason.com/blog/gandcrab-evasive-infection-chain>>. [Online; accessed 07-November-2019]. Citado na página 18.
- DEKKER, R. The importance of having data-sets. *27th IATUL Conference, 22 - 25 May 2006*, 01 2006. Citado na página 8.
- DHILLON, G.; BACKHOUSE, J. Security Management in the New Millennium. v. 43, n. 7, p. 125–128, 2000. Citado 3 vezes nas páginas 7, 10 e 11.
- DU, M. Cisco 2017 Annual Cybersecurity Report. *Journal of World Trade*, v. 50, n. 4, p. 675–704, 2016. ISSN 10116702. Citado na página 7.
- FEDERAL INFORMATION PROCESSING STANDARDS. *FIPS PUB 199: Standards for security categorization of federal information and information systems*. Gaithersburg, 2004. 13 p. Citado na página 10.
- FRANCO, E.; PAGANI, E. Sistema de Detecção de Intrusão - Artigo Revista Infra Magazine 1. 2016. Citado na página 13.
- GAZET, A. Comparative analysis of various ransomware virii. p. 77–90, 2010. Citado na página 11.
- GUPTA, A. et al. p. 1–10, 2009. Citado na página 8.

- KEANE, J. *TeslaCrypt ransomware grows as victims pay up*. 2015. <<https://www.digitaltrends.com/computing/teslacrypt-ransomware-grows-as-victims-pay-up/>>. [Online; accessed 05-December-2019]. Citado na página 9.
- KHARAZ, A. et al. UNVEIL : A Large-Scale , Automated Approach to Detecting Ransomware This paper is included in the Proceedings of the. 2016. Disponível em: <<https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/kharaz>>. Citado 2 vezes nas páginas 7 e 14.
- KOLBITSCH, C. et al. Effective and Efficient Malware Detection at the End Host. 2009. Citado na página 13.
- KURNIAWAN, A.; RIADI, I. Detection and Analysis Cerber Ransomware Based on Network Forensics Behavior. v. 20, n. 5, p. 836–843, 2018. Citado na página 18.
- LOPES, T. et al. Aplicação do algoritmo Random Forest como classificador de padrões de falhas em rolamentos de motores de indução. 2017. Citado na página 35.
- MUNHOZ, V. *WannaCry, o ransomware que fez o mundo chorar na sexta-feira (12)*. 2017. <<https://www.tecmundo.com.br/malware/116652-wannacry-ransomware-o-mundo-chorar-sexta-feira-12.htm>>. [Online; accessed 05-December-2019]. Citado 2 vezes nas páginas 3 e 12.
- NORDSTRÖM, T.; SÖDERSTRÖM, M.; HANSETH, O. Business Development in IT-dependent organisations. n. c, 2000. Citado na página 7.
- OLENICK, D. *Microsoft Office 365 hit with massive Cerber ransomware attack, report*. 2016. <<https://www.scmagazineuk.com/microsoft-office-365-hit-massive-cerber-ransomware-attack-report/article/1477835>>. [Online; accessed 05-December-2019]. Citado na página 9.
- OLENICK, D. *GandCrab ransomware strikes Doctors' Management Services*. 2019. <<https://www.scmagazine.com/home/security-news/ransomware/gandcrab-ransomware-strikes-doctors-management-services/>>. [Online; accessed 05-December-2019]. Citado na página 9.
- Ponemon Institute LLC, Accenture. *The Cost Of Cybercrime*. [S.l.], 2019. Citado na página 7.
- PRODUCTS, T. N. S. *Cerber ransomware*. 2017. <<https://temasoft.com/information/cerber-ransomware-analysis/>>. [Online; accessed 10-November-2019]. Citado na página 30.
- RICHARDSON, R.; NORTH, M. Ransomware: Evolution, mitigation and prevention. 01 2017. Citado na página 12.
- RING, M. et al. A Survey of Network-based Intrusion Detection Data Sets. p. 1–17, 2019. Citado na página 8.
- SANTANA, R. *Café Com Código 09: Entendendo Métricas de Avaliação de Modelos*. 2017. <<https://minerandodados.com.br/cafe-com-codigo-09-metricas-de-avaliacao-de-modelos/>>. [Online; accessed 04-December-2019]. Citado na página 36.

- Scaife, N. et al. Cryptolock (and drop it): Stopping ransomware attacks on user data. p. 303–312, June 2016. ISSN 1063-6927. Citado na página 8.
- SCHAPIRE, R. *Theoretical Machine Learning*. 2008. University Lecture. Citado na página 35.
- SHARAFALDIN, I.; LASHKARI, A. H.; GHORBANI, A. A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. n. Cic, p. 108–116, 2018. Citado na página 15.
- SHIRAVI, A. et al. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers and Security*, Elsevier Ltd, v. 31, n. 3, p. 357–374, 2012. ISSN 01674048. Disponível em: <<http://dx.doi.org/10.1016/j.cose.2011.12.012>>. Citado 2 vezes nas páginas 14 e 15.
- SOO, Z.; NG, N.; CHEN, S. *Tens of thousands of Chinese firms, institutes affected in WannaCry global cyberattack*. 2017. <<https://www.scmp.com/news/china/policies-politics/article/2094377/tens-thousands-chinese-firms-institutes-affected>>. [Online; accessed 05-December-2019]. Citado na página 9.
- Sophia Wang. *Analysis of the TeslaCrypt Family*. [S.l.], 2017. Citado na página 19.
- STALLINGS, W. *Cryptography and Network Security: Principles and Practice*. 6th. ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2013. ISBN 0133354695, 9780133354690. Citado na página 11.
- TAHIR, R. A Study on Malware and Malware Detection Techniques. n. March, p. 20–30, 2018. Citado na página 13.
- TREND LABS. *Unseen Threats , Imminent Losses*. [S.l.], 2018. Citado na página 7.
- VERBERT, K. et al. Dataset-Driven Research to Support Learning and Knowledge Analytics. v. 15, p. 133–148, 2012. Citado na página 8.
- VIEGAS, E. K. Detecção de intrusão baseada em anomalia para ambientes de produção. 2016. Citado na página 13.
- YAMBAO, F. A. M. *Businesses as Ransomware’s Goldmine: How Cerber Encrypts Database Files*. 2016. <<https://blog.trendmicro.com/trendlabs-security-intelligence/how-cerber-encrypts-database-files/>>. [Online; accessed 10-November-2019]. Citado na página 30.
- YOUNG, A.; YUNG, M. Cryptovirology: Extortion-based security threats and countermeasures. 09 1996. Citado na página 12.