São Paulo, Brazil
October 29-30, 2018

The Tenth International Conference on
FORENSIC COMPUTER SCIENCE and CYBER LAW

www.ICoFCS.org

DOI: 10.5769/C2018008 or http://dx.doi.org/10.5769/C2018008

Uma Proposta de Abordagem para Análise Forense de Sistemas Invadidos por *Ransomware*

Wilson Leite da Silva Filho[1]

(1) Instituto Geral de Perícias (IGP/SC), Email: wleitefilho@gmail.com

Resumo: A perícia em casos de invasão de sistemas representa um desafio ao especialista, principalmente pela diversidade de tecnologias envolvidas e a grande quantidade de dados que precisa ser analisada. O objetivo do trabalho é apresentar uma abordagem desse problema realizada em um caso real de sistema invadido vítima de *ransomware*. O trabalho descreve o cenário do sistema invadido, os processos usados para obtenção dos dados, a necessidade de informações adicionais para análise dos dados obtidos e como a abordagem utilizada pôde contribuir para as conclusões da perícia. Ao final, são discutidas potenciais aplicações da abordagem em casos semelhantes e trabalhos futuros.

Palavras-chave: *ransomware*, invasão de sistemas, computação forense, perícia computacional forense.

Abstract: The forensic analysis in computer invasion cases are a challenger for the expert, due to the technology diversity and huge amount of data to be analyzed. The objective of this paper is to present an approach of this problem that was used in a real case of computer invasion by ransomware. The paper describes the scenario of the invaded system, the process used to get the data, the necessity of additional information to analyze the obtained data and how this approach can help the forensics conclusion. In the end, the possible application of this approach and future work are discussed.

Key words: *ransomware*, system hacking, computer forensics.

I. Introdução

A análise forense de sistemas que foram alvo de ataques ou invasão apresenta um grande desafio ao especialista forense. A diversidade de plataformas, sistemas e softwares em execução exige do perito um amplo conhecimento sobre diversas tecnologias envolvidas no processo. Investigar esse processo com base apenas nos rastros digitais que ficaram registrados em dispositivo de memória secundária aumenta a dificuldade da análise. Este é o cenário típico das análises post-mortem.

Existem etapas básicas que devem ser seguidas em uma perícia forense digital: identificação, coleta, preservação, análise e apresentação. Na coleta, obtém-se os dados digitais de forma que não os altere ou que os altere o mínimo possível.

Na preservação, é comum se fazer uma cópia bit a bit da evidência original e garantir a integridade dos dados por meio de funções *hash* [3]. Muitas vezes os primeiros atendentes de um evento de segurança da informação não tomam todas as medidas necessárias para preservação da prova digital. Mandia, Pepe e Luttegens apontam que geralmente as ocorrências de segurança da informação são inicialmente tratadas pela equipe de suporte técnico local e não por uma equipe especializada em invasão de sistema ou forense computacional [5]. Diferentemente de uma equipe especializada, o foco do suporte técnico pode ser o de apenas restaurar o funcionamento do sistema, sem atentar para os processos de preservação necessários para uma boa análise forense. Perdida a oportunidade de salvar as informações voláteis nessa fase, restará apenas o conteúdo da memória secundária para ser analisado pelo investigador digital. Portanto, apesar de não ser o mais indicado em casos de invasão de computadores, em vários casos, a perícia post-mortem é a única opção que se resta.

Desse modo, o objetivo do trabalho é apresentar uma abordagem de análise forense utilizada em um caso real de sistema invadido por *ransomware*, em que os dados disponíveis limitavam-se ao acesso ao disco rígido do servidor alvo. A motivação inicial surgiu da necessidade de se realizar essa perícia e da constatação que este tipo de análise está se tornando mais frequente na instituição.

O restante do artigo está organizado da seguinte forma: na seção dois, é realizada uma revisão da literatura e de trabalhos relacionados e a diferença da presente abordagem em relação a esses trabalhos. A seção três apresenta a abordagem de forma esquematizada, dividindo-a em etapas. Na quatro, são apresentados os dados obtidos em cada etapa e as descobertas realizadas com a análise desses dados. Por fim, a seção cinco conclui o trabalho, discute limitações e problemas encontrados e possíveis trabalhos futuros.

II. Trabalhos relacionados

Para se periciar equipamentos que foram vítimas de invasão, é fundamental que se obtenha dados de diversas fontes. Carvey nos diz que o Registro é uma rica fonte de dados para administradores de sistema e investigadores forenses [2]. Segundo o autor, em muitos casos, softwares usados por invasores deixarão rastros no Registro, permitindo que o expert possa encontrar vestígios sobre o incidente. O conteúdo do Registro pode ser lido pelo aplicativo RegEdit do Windows ou algum outro software de terceiros. Para fins de investigação forense, existe uma ferramenta denominada RegRipper (https://github.com/ keydet89/ RegRipper2.8) que extrai diversas informações de interesse investigativo.

Seguindo nessa mesma linha, a análise dos logs é fundamental. No Windows, existe um mecanismo de log comum do sistema e de alguns outros componentes denominado Log de Eventos. Ele registra uma variedade de eventos do dia a dia do Windows. É dividido em categorias, tais como eventos de aplicação, de segurança, do sistema, encaminhados etc. [2].

Até aqui, foram apresentadas algumas fontes de informações sobre o sistema. Porém, esse número pode ser muito maior. Analisar todas elas individualmente e tentar fazer correspondência de eventos dessa forma será mais complicado do que analisar um log único, com todos os dados das diversas fontes reunidos. Além disso, levar em conta a ordem cronológica (linha do tempo) dos eventos é fundamental. A técnica que permite agregar todos os logs, respeitando-se a ordem cronológica é denominada super linha do tempo. A ferramenta supertimeline/plaso permite realizar essa tarefa (https://github.com/log2timeline/plaso /wiki). Ao ser executada, ela lê o conteúdo do disco, interpretando os tipos de arquivos que entende e agrega todo o conteúdo em um grande arquivo de log. Entre os tipos de arquivos interpretados estão: arquivos de cache e histórico dos navegadores Web, *journal* do NTFS, arquivos pcap, logs de eventos, firewall, lixeira, registro do Windows, entre outros.

Detecção e análise de *malware* é uma outra atividade da computação forense. A Cartilha de Segurança para Internet da CERT.br define *malware* como programas especificamente desenvolvidos para executar ações danosas e atividades maliciosas em um computador [1]. Velho et al cita três cenários relacionados a softwares maliciosos de interesse pericial: *malware* como sofware suspeito, em que um software conhecido dos usuários está sendo usado, mas suspeita-se que ele tenha funcionalidades secundárias maliciosas; ataques

utilizando *malwares*, onde já há indícios da ocorrência de um ataque a um dispositivo computacional, devendo-se buscar vestígios que determinem o objetivo, a materialidade e, se possível, a autoria do ataque; *malware* como elemento secundário, onde o software malicioso não é o foco principal do exame pericial, mas atua como um provedor adicional de informações. Sobre os tipos de análise, esses mesmos autores classificam como: análise estática, que engloba todas as formas possíveis para obtenção de informações acerca do funcionamento do *malware*, entretanto, sem executá-lo; análise dinâmica, que é o estudo do funcionamento do software com o programa em execução; análise post-mortem, que é a análise de um *malware* com base nos vestígios deixados por este após sua execução. Ao se encontrar *malwares* em um computador, não necessariamente significa que ele foi executado. Há casos em que os arquivos de *malware* estão presentes, mas por algum motivo, nunca se tornaram ativos. Uma análise nos vestígios do sistema operacional pode indicar a efetiva ação do software malicioso [6]. Sainju e Atkison realizaram diversos testes com *malwares* conhecidos e verificaram a importância dos registros dos logs de eventos do Windows para este tipo de análise [11].

Uma outra área correlata à computação forense, importante nesse trabalho, é a de testes de invasão. É útil ao periciar um sistema, conhecer as técnicas do adversário. Weidman define testes de invasão como a simulação de ataques reais para avaliar os riscos associados a potenciais brechas de segurança [7]. Além da identificação, as vulnerabilidades encontradas são exploradas sempre que possível, para avaliar o que os invasores poderiam obter após uma exploração bem-sucedida das falhas. As fases de um teste de invasão, segundo a autora são: preparação, coleta de informações, modelagem das ameaças, análise de vulnerabilidades, exploração de falhas e pós-exploração de falhas.

Zhang et al discutem sobre a importância dos logs de diversas fontes na investigação forense de invasões de sistemas e da dificuldade de lidar com essa quantidade de dados. Nesse estudo os autores propõem uma técnica de filtragem baseada em padrões para redução da quantidade de logs inúteis [10].

Yoan, Bertaux, e Tahar apresentam uma proposta baseada em semântica, implementada por meio de ontologias, para lidar com o problema da análise de logs de um sistema. A proposta consiste em fornecer ao analista uma visão de conhecimento dos eventos do sistema com um nível de agrupamento e abstração maior do que os dados brutos [9].

O presente trabalho lida com o problema da grande quantidade de logs com uma abordagem diferente da usada nos dois últimos trabalhos citados. A ideia aqui é subsidiar o especialista com informações de diversas fontes, por meio de várias técnicas, para permitir um olhar mais seletivo e direcionado aos dados coletados do sistema. A vantagem sobre os trabalhos anteriores é que essa abordagem pode representar um atalho para a compreensão dos grandes arquivos de logs, uma vez que o especialista, ao chegar na fase de análise dos logs em si, já contará com conhecimento acerca de diversos aspectos importantes do sistema invadido. Apesar de ser diferente, a presente abordagem não é incompatível ou excludente às técnicas usadas por Zhang et al e Yoan, Bertaux, e Tahar [9] [10].

III. Descrição da Abordagem do Problema

Essa seção do artigo apresenta, de forma esquematizada, a abordagem realizada no caso real periciado. Ressalta-se que antes das etapas ilustradas aqui, os procedimentos básicos de forense computacional (identificação, coleta, preservação e indexação dos arquivos) já haviam sido executados. Os esquemas estão ilustrados nas figuras 1 a 4.

A abordagem foi dividida em sete etapas. A etapa 1 consistiu em obter dados de configuração do sistema invadido. É uma fase de reconhecimento. Ao final da fase, tem-se um relatório com diversas características de versão e configuração do sistema. A etapa de reconhecimento de um sistema é comum em testes de invasão, mas diferente deste método, que tenta fazer o reconhecimento a partir de um ambiente externo, aqui usamos os arquivos disponíveis do próprio sistema, já que temos acesso direto a eles.

Na etapa 2 ocorreu a busca por *malwares* que estariam no disco do sistema invadido. Processou-se a evidência com um antivírus, extraiu-se os

arquivos apontados como *malwares* e submeteu-os a uma suíte de antivírus para se obter um resultado mais detalhado. Uma espécie de segunda opinião sobre a classificação dos *malwares*.

Na etapa 3, buscou-se identificar uma vulnerabilidade muito comum nos mais variados tipos de sistema. O uso incorreto de senhas de acesso, por exemplo, é uma vulnerabilidade muito visada por atacantes. O processo consiste em extrair do sistema o arquivo de armazenamento de *hashes* das senhas e submetê-lo a algum processo de quebra de senhas. Senhas frágeis não resistirão ao processo de quebra, mostrando uma importante vulnerabilidade e possível caminho que pode ter sido explorado na invasão.

Na quarta etapa, foram usadas as técnicas dos métodos de teste de invasão para identificar vulnerabilidades que estão expostas a um agente externo.

Na etapa 5, foi gerada a super linha do tempo, agregando-se todos os logs do sistema. Foi na super linha do tempo que boa parte das informações que permitem determinar a dinâmica da invasão estavam armazenadas.

Na etapa 6 que o trabalho árduo de análise dos diversos eventos do sistema, provenientes de diversas fontes foi realizado. Foi com esses dados da super linha do tempo que se conseguiu entender os passos que levaram a uma invasão bem-sucedida do sistema. Porém, partir direto para a análise dessa linha do tempo teria sido contraprodutivo. A super linha do tempo é um arquivo muito grande, que pode chegar a centenas de milhares ou até mesmo milhões de linhas de logs e dados, das mais variadas fontes. Ter informações sobre o sistema, suas configurações e fraquezas e *malwares* envolvidos permitiu um olhar mais seletivo nos logs de linha do tempo. Dessa forma, as informações obtidas nas etapas de 1 a 4 auxiliaram na compreensão do grande número de eventos de sistema presentes nessa etapa.

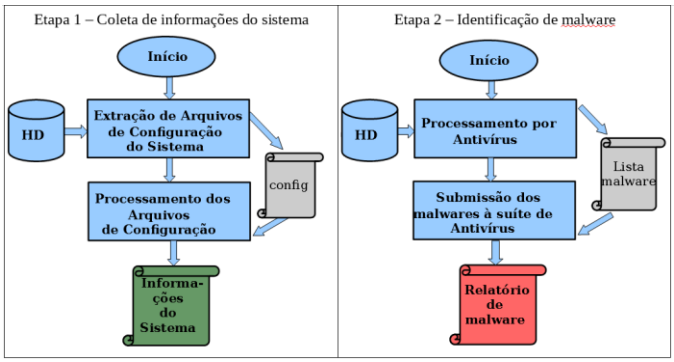

Figura 1. Esquematização das etapas 1 e 2 da abordagem utilizada.

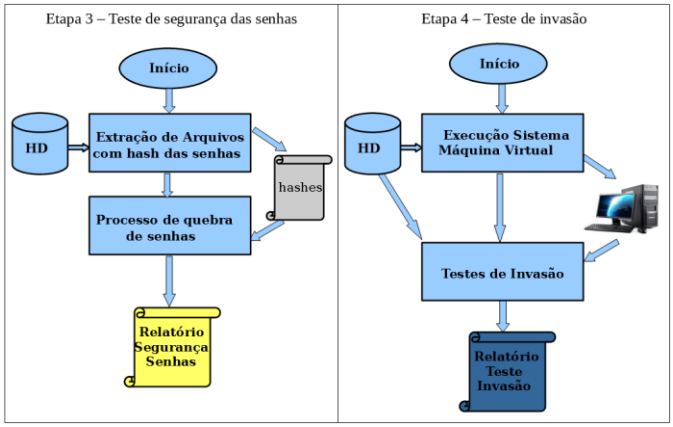

Figura 2. Esquematização das etapas 3 e 4 da abordagem utilizada.

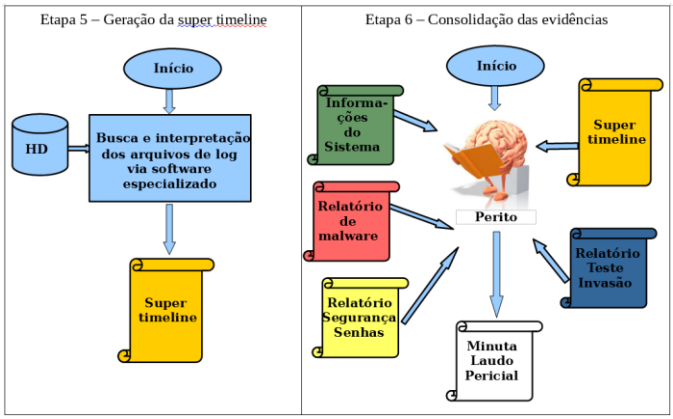

Figura 3. Esquematização das etapas 5 e 6 da abordagem utilizada.

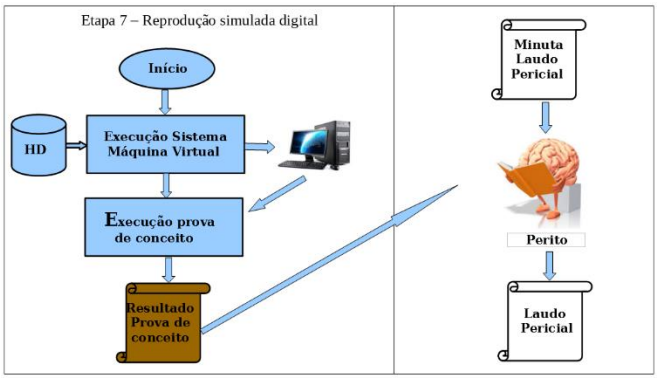

Figura 4. Esquematização da etapa 7 da abordagem utilizada.

IV. Descrição da Abordagem do Problema

Nessa seção são apresentadas as técnicas, recursos e dados obtidos com o emprego da abordagem da seção anterior. Na etapa 1 (coleta de informações do sistema), montou-se a imagem do disco rígido do sistema invadido usando software FTKImager e foi feita a extração dos arquivos de Registro do Windows. Os arquivos de registro foram submetidos a ferramenta RegRipper para interpretação e extração dos dados de interesse. Dessa primeira fase foram obtidas informações de configuração, tais como: o sistema operacional (Windows 6.3 - 2012 – Server); o endereço IP (192.168.1.11); a lista de usuários do sistema; constatação de que o Windows Firewall estava desabilitado e o serviço RemoteDesktop estava ativo. Na etapa 2, foi executado antivírus sobre a imagem montada da evidência. Os arquivos detectados foram extraídos e submetidos ao site VirusTotal (https://www. virustotal.com), que analisa os *malwares* utilizando softwares de antivírus de diversos fabricantes. A tabela 1 resume o resultado dessa fase.

Nome	Taxa detecção	Descrição
NLBrute 1.2 x64.exe	46/60	HackTool/Win32.BruteForce.C1767822;
		Trojan..Zbot.frh; HackTool.Agent
Baixaki_winrar.exe	22/60	Win32:UnwantedSig; not-a-virus:AdWare.Win32.DealPly.avsuk
KPortScan 3.exe	23/60	HackTool.Win32.Kscan.a; HKTL_PORTSCAN
SECOH-QAD.exe	24/60	not-a-virus:RiskTool.Win64.ProcPatcher.a; Hacktool
xRdp.v2.1.exe	34/61	HackTool.Patcher; HEUR:HackTool.Win32.RpdPatch.gen

Tabela 1. *Malwares* encontrados.

Na terceira etapa, foi testada a qualidade das senhas dos usuários do Windows. Os hashes das senhas foram extraídos do arquivo SAM do sistema e submetidos a ferramenta hashcat, em uma máquina com placas GPUs. Três usuários tiveram suas senhas rapidamente descobertas (administrador, admin10 e tesouraria). As senhas utilizadas possuíam quantidade de caracteres insuficientes para boa segurança e parte da sua formação usava palavras constantes em dicionários, o que permitiu sua descoberta rapidamente.

Na quarta etapa, foi realizado o teste de invasão. A imagem do computador periciado foi configurada e executada numa máquina virtual por meio da VirtualBox da Oracle (https://www.virtualbox.org/). O ambiente de rede foi configurado para acessar rede virtual exclusiva às máquinas virtuais. Nenhum acesso à rede corporativa ou à Internet foi permitido. Em uma segunda máquina virtual, foi usada a distribuição Kali Linux para realização dos testes de invasão (https://www.kali.org/). Além das ferramentas de segurança já presentes na Kali, foi instalado o scanner de vulnerabilidades Nessus (https://www.tenable.com/products/nessus-vulnerability-scanner). Foram escaneadas as portas abertas por meio do nmap e usado o Nessus para identificação de vulnerabilidades.

Entre as vulnerabilidades encontradas foram listadas: MS14-066 (Vulnerability in Schannel), MS16-047 (Security update for SAM and LSAD), Windows Terminal Services Enabled.

Na etapa 5, foi usada a ferramenta supertimeline/plaso para processar todo o conteúdo da imagem do servidor. Foi escolhido como formato de saída um arquivo tipo CSV. O arquivo gerado tinha 933.744 linhas de log e 604 MB de tamanho.

Na etapa 6, foram analisados todos os dados disponíveis, inclusive a super linha do tempo, com quase um milhão de linhas de log. A interpretação dos dados das fases anteriores permitiu ter em mãos as seguintes informações: uma lista de *malware*, algumas vulnerabilidades e as configurações do servidor. Era de conhecimento também a data de ocorrência dos fatos. Primeiramente, procurou-se nos logs algum registro dos nomes dos arquivos de *malware*. A resposta foi positiva para o *malware* "NLBrute.exe", conforme trecho de log a seguir (figura 5).

date	time	source	sourcetype	filename
05/11/2017	11:29:10	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Security.evtx
05/11/2017	11:29:15	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Security.evtx
05/11/2017	11:29:22	FILE	NTFS_DETECT atime	TSK:/Users/ds/Desktop/NLBrute.exe
05/11/2017	11:29:32	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Microsoft-Windows-ServerMar
05/11/2017	11:29:46	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Security.evtx
05/11/2017	11:29:50	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Security.evtx
05/11/2017	11:30:02	FILE	NTFS_DETECT ctime	TSK:/Users/ds/Desktop/NLBrute.exe
05/11/2017	11:30:06	REG	NTUSER key	TSK:/Users/ds/NTUSER.DAT
05/11/2017	11:30:06	REG	UNKNOWN key	TSK:/Windows/AppCompat/Programs/Amcache.hve
05/11/2017	11:30:06	REG	UNKNOWN key	TSK:/Windows/AppCompat/Programs/Amcache.hve

Figura 5. Registro do malware NLBrute.exe.

Com o conhecimento de que havia senhas inadequadas, firewall desativado e um *malware* de força bruta identificado como efetivamente usado, levantou-se a hipótese de um ataque ao serviço de Remote Desktop do Windows.

Buscou-se por evidências de ataque de força bruta na super linha do tempo. Foram identificadas várias tentativas fracassadas de acesso remoto via protocolo Remote Desktop. Um pequeno trecho desse ataque é ilustrado na figura 6. Tentativas com sucesso de conexão via Remote Desktop foram registradas em vários momentos (figura 7).

date	time	source	sourcetype	extra
05/11/2017	11:34:33	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-
05/11/2017	11:35:06	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-
05/11/2017	11:35:09	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-
05/11/2017	11:35:38	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-
05/11/2017	11:35:41	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-
05/11/2017	11:36:14	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-
05/11/2017	11:36:18	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">-

Figura 6. Ataques ao Remote Desktop.

date	time	source	sourcetype	extra
05/11/2017	18:15:57	EVT	WinEVTX	<Data Name="TargetUserSid">S-1-5-7</Data>- <Data Name="TargetUserName">LOGON ANÃ"NI
05/11/2017	18:15:57	EVT	WinEVTX	<Data Name="TargetUserSid">S-1-5-7</Data>- <Data Name="TargetUserName">LOGON ANÃ"NI
05/11/2017	18:15:57	EVT	WinEVTX	xml_string: <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">- <System>- <E

Figura 7. Conexão remota via Remote Desktop.

Outro teste foi comparar a lista de usuários obtidos na etapa 2 e os usuários que apareciam disponíveis na tela de logon Verificou-se a ausência do usuário Admin10 na tela de logon. Procurando por esse usuário na super linha do tempo, pôde-se ver que sua criação ocorreu no período que atividades suspeitas já haviam começado (figura 8). Após a criação do Admin10, essa conta foi usada diversas vezes para acesso remoto. Posteriormente, foi identificada a ocorrência dos primeiros arquivos criptografados (figura 9) e a criação do arquivo com a mensagem deixada pelo atacante (figura 10).

date	time	source	sourcetype	desc
05/11/2017	23:17:43	REG	SAM key	[\SAM\Domains\Account\Users\Names\Admin10] (default): [UNKNOWN
05/11/2017	23:17:43	EVT	WinEVTX	[4728 / 0x1278] Record Number: 97666 Event Level: 0 Source Name: M
05/11/2017	23:17:43	REG	SAM key User Account Information	[\SAM\Domains\Account\Users] account_rid: 1007 full_name: Admin10
05/11/2017	23:17:43	EVT	WinEVTX	[4720 / 0x1270] Record Number: 97667 Event Level: 0 Source Name: M

Figura 8. Criação do usuário Admin10.

date	time	source	sourcetype	filename
05/13/2017	00:02:11	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Security.evtx
05/13/2017	00:02:11	FILE	NTFS_DETECT ctime	TSK:/backup/SAPO/Mes02/Semana02/20170205/sapo.db.aac
05/13/2017	00:02:12	FILE	NTFS_DETECT atime	TSK:/backup/SAPO/Mes02/Semana02/20170205/sapo.log.aac
05/13/2017	00:02:15	EVT	WinEVTX	TSK:/Windows/System32/winevt/Logs/Security.evtx

Figura 9. Primeiros arquivos criptografados pelo *ransomware*.

date	time	source	sourcetype	filename
05/13/2017	00:10:08	FILE	NTFS_DETECT ctime	TSK:/backup/SAPO/Mes02/Semana02/20170206/sapo.log.aac
05/13/2017	00:10:08	FILE	NTFS_DETECT atime	TSK:/backup/SAPO/Mes02/Semana02/20170206/Saiba como recuperar seus arquivos.txt
05/13/2017	00:10:08	FILE	NTFS_DETECT atime	SERVER

Figura 10. Criação do arquivo com as instruções para pagamento resgate.

date	time	source	desc
05/13/2017	13:28:58	WEBHIS	Administrador@https://www.bleepingcomputer.com/download/malwarebytes-anti-ransomware/ Access cou
05/13/2017	13:29:01	EVT	[4625 / 0x1211] Record Number: 105587 Event Level: 0 Source Name: Microsoft-Windows-Security-Auditi
05/13/2017	13:29:46	EVT	[4625 / 0x1211] Record Number: 105588 Event Level: 0 Source Name: Microsoft-Windows-Security-Auditi
05/13/2017	13:29:53	EVT	[30800 / 0x7850] Record Number: 51433 Event Level: 2 Source Name: Microsoft-Windows-SMBClient Co

Figura 11. Instalação do *antiransomware*.

Por último, foi executada a etapa 7. Em uma terceira máquina virtual com Windows 7 foi copiado o *malware* "NLBrute.exe", realizada a sua

configuração e iniciado um ataque contra a máquina periciada. O *malware* conseguiu descobrir a senha do usuário Administrador via protocolo Remote Desktop. Analisando-se os logs desse ataque, pôde-se observar que as assinaturas de logs desse ataque eram semelhantes às existentes em dias anteriores à criptografia dos arquivos, indicando que havia, pelo menos, uma outra máquina na rede comprometida e que esse ataque foi realizado a partir dela para se obter o controle da máquina periciada. O programa que realizou a criptografia não foi encontrado e serviços on-line *antiransomware* não conseguiram reverter a cifragem. As vulnerabilidades, o tipo e a dinâmica do ataque que permitiram controle da máquina periciada ficaram esclarecidos.

V. Conclusão

A perícia em dispositivos que foram alvo de ataques e invasões cibernéticas é uma tarefa desafiadora. A abordagem utilizada apresentou um caminho lógico de tarefas que contribuíram com os resultados da perícia, auxiliando na eficiência da análise de todo conjunto de evidências, na formulação e teste de hipóteses e na elaboração do laudo.

Uma dificuldade encontrada, mesmo com as informações das etapas anteriores, foi navegar e relacionar eventos da super linha do tempo sem uma ferramenta específica para esse fim. Associar uma ferramenta específica à etapa 6 pode melhorar o desempenho das análises.

Trabalhos futuros podem testar o uso de ferramentas para auxílio nas pesquisas dos logs da super linha do tempo. Ferramentas como a pilha ELK (ElasticSearch, Logstash e Kirbana) podem contribuir com a manipulação dos grandes arquivos de log (https://www.elastic.co/webinars/introduction-elk-stack). Também como trabalho futuro, pode-se usar a mesma abordagem em casos semelhantes ou que envolvam outros sistemas, como Linux e talvez até sistemas de dispositivos móveis (Android e iOS). O uso da abordagem em outros casos permitiria seu aprimoramento com base em novas demandas que estes casos possuiriam, aumentando a sua generalização, com potencial para transformá-la em um método.

Referências

[1] Cartilha de Segurança para Internet – CERT.br; https://cartilha.cert.br/malware/, acesso em 08/09/2017

[2] Carvey, Harlan – Windows Forensic Analysis – USA: Syngress, 2009

[3] Grande, Carlos Lopez; Guadron, Ricardo Salvador - Computer Forensics - when crime makes use of technology – 36th Central American and Panama Convention (CONCAPAN XXXVI), IEEE, 2016

[4] Log2Timeline/Plaso; https://github.com/log2timeline/plaso/wiki, acesso em 07/09/2017

[5] Mandia, K; Pepe, M; Luttegens, J; Incident Response & Computer Forensics; Third Edition; MacGraw-Hill Education, 2014

[6] Velho, Jesus Antonio; et al – Tratado de Computação Forense – Millenium Editora; São Paulo, 2016

[7] Weidman, Georgia; Teste de Invasão – Uma introdução ao hacking; Novatec Editora; São Paulo, 2014

[8] Windows Sysinternals; https://docs.microsoft.com/en-us/sysinternals/, acesso em 09/08/2017

[9] Yoan Chabot, Aur´elie; Bertaux, Christophe Nicolle and Tahar Kechadi - Automatic Timeline Construction and Analysis for Computer Forensics Purposes - IEEE Joint Intelligence and Security Informatics Conference, 2014

[10] Zhang, Jian; et al - A Method to Automatically Filter Log Evidences for Intrusion Forensics - 33rd International Conference on Distributed Computing Systems Workshops, IEEE, 2013

[11] Sainju, Arpan Man; Atkison, Travis - An Experimental Analysis of Windows Log Events Triggered by Malware - April 2017 ACM SE '17: Proceedings of the SouthEast Conference, 2017