HAMILTON FONTE II

DIZANG: UMA SOLUÇÃO PARA COLETA DE EVIDÊNCIAS FORENSES DE ATAQUES DE INJEÇÃO NA NUVEM

Documento apresentado à Escola Politécnica da Universidade de São Paulo para a realização do Exame de Qualificação de Mestrado em Engenharia Elétrica.

São Paulo
2017

HAMILTON FONTE II

DIZANG: UMA SOLUÇÃO PARA COLETA DE EVIDÊNCIAS FORENSES DE ATAQUES DE INJEÇÃO NA NUVEM

Documento apresentado à Escola Politécnica da Universidade de São Paulo para a realização do Exame de Qualificação de Mestrado em Engenharia Elétrica.

Área de concentração:
Engenharia da Computação

Orientador:
Marcos Antonio Simplicio Junior

São Paulo
2017

RESUMO

Arquiteturas em nuvem são cada vez mais comuns, e também o número de problemas de segurança envolvendo essa tecnologia. Infelizmente, devido à natureza volátil de recursos na nuvem, a coleta de evidências para análise forense nesse ambiente tem esbarrado em desafios práticos e legais. Este trabalho analisa propostas voltadas a resolver os desafios existentes na coleta evidências na nuvem, discute suas limitações, e então apresenta uma solução visando suplantá-las. Especificamente, a proposta tem como foco a reprodutibilidade do processo de coleta, sem com isso violar jurisdições ou a privacidade dos não envolvidos na investigação.

ABSTRACT

Cloud architectures are becoming more common, and so are the number of security issues surrounding this technology. Unfortunately, due to the volatile nature of cloud resources, the task of gathering evidence for forensic analysis in this environment runs into practical and legal challenges. This paper analyzes proposals aimed at addressing existing challenges when collecting evidence in the cloud, discuss their limitations, and then presents a solution to overcome them. Specifically, the proposal focuses on the reproducibility of the collection process, without violating jurisdictions or the privacy of those not involved in the investigation.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

VM	*Virtual Machines* - Máquina Virtual
SO	Sistema Operacional
SaaS	*Software as a Service* - Programa como serviço
PaaS	*Platform as a Service* - Plataforma como serviço
IaaS	*Infrastructure as a Service* - Infra-estrutura como serviço
LXC	*Linux Conteiners* - Contêiners Linux

LIST OF SYMBOLS

$a >> b$	Símbolo 1
$\|\|$	Símbolo 2
$\|x\|$	Símbolo 3
$\oplus$	Símbolo 4
$gcd(a, b)$	Símbolo 5
$\mathcal{H}$	Símbolo 6
$\mathcal{O}$	Símbolo 7

CONTENTS

1 INTRODUÇÃO

1.1 Problema de pesquisa

Técnicas de virtualização, replicação de serviços e compartilhamento de recursos entre múltiplos usuários (multi-inquilinato) proveem a nuvens computacionais uma elevada escalabilidade (MORSY; GRUNDY; MULLER, 2010). Ao mesmo tempo, tais mecanismos também criam uma elevada volatilidade dos recursos virtuais que executam aplicações em nuvem. Afinal, quando submetida a uma carga elevada, uma aplicação hospedada na nuvem pode criar clones das máquinas virtuais (*virtual machines* – VMs) que a hospedam e balancear a carga entre elas, de modo a atender à demanda sem prejuízos na qualidade do serviço oferecido. Após esse pico, as máquinas que foram clonadas são normalmente desativadas, seus recursos liberados e o sistema retorna à capacidade anterior, evitando-se custos desnecessários.

Embora interessante do ponto de vista de eficiência e custos, do ponto de vista forense a volatilidade da nuvem traz problemas em caso de ataques. Por exemplo, caso uma das instâncias de processamento virtuais criadas temporariamente seja alvo de ameaças que atuam diretamente na sua memória, sem deixar rastros em discos (e.g., arquivos de *log*), as evidências desse evento podem ser completamente perdidas após elas serem desativadas e terem seus recursos liberados. Essa dificuldade é ainda agravada por aspectos como multi-inquilinato e multi-jurisdição típicas de soluções em nuvem (GILBERT; SUJEET, 2008). Especificamente, o aspecto multi-inquilino dificulta a obtenção do *hardware* que executa as aplicações de interesse, pois, como

ele é compartilhado por vários usuários, removê-los para análise poderia levar a uma violação de privacidade dos usuários não relacionados à investigação. Já a natureza distribuída da nuvem pode levar à alocação de informações relevantes à investigação em vários países, dificultando a obtenção das mesmas em especial quando não existem acordos de cooperação entre as entidades envolvidas (DYKSTRA; SHERMAN, 2012). Combinadas, tais características dificultam a coleta de evidências com a credibilidade necessária para que elas possam ser usadas em processos legais, o que exige o respeito à privacidade, à jurisdição e à cadeia de custódia, bem como a reprodutibilidade do processo de coleta (RAHMAN; KHAN, 2015).

Embora existam soluções na literatura que abordam a coleta de informações de nuvem com o propósito de análise forense, a grande maioria delas aborda a coleta, o transporte e o armazenamento de forma isolada. Por exemplo, trabalhos como (DYKSTRA; SHERMAN, 2013) e (REICHERT; RICHARDS; YOSHIGOE, 2015) tratam de fatores como multi-inquilinato e multi-jurisdição, discutindo formas de coleta e preservação da evidência fora da nuvem. Já estudos como (GEORGE; VENTER; THOMAS, 2012) se concentram na análise forense para a coleta de evidência de máquinas virtuais enquanto elas estão em execução, enquanto trabalhos como (SANG, 2013) abordam a questão de processos de garantia de cadeia de custódia em ambientes de nuvem para transporte da evidência. Por outro lado, não foram identificadas na literatura propostas que (1) descrevam como o dado é coletado e armazenado observando a cadeia de custódia, e (2) visem garantir que, mesmo que um recurso virtualizado (e.g., uma VM) seja desalocada, haja condições de se reproduzir o processo de coleta de evidências.

1.2 Objetivos

O presente trabalho visa suplantar tais limitações por meio de uma proposta que tem como focos (1) a reprodutibilidade do processo de coleta, (2) o estabelecimento de vínculo entre a evidência coletada e sua origem, (3) a preservação da jurisdição e

da privacidade dos não envolvidos na investigação e (4) a garantia de custódia da evidência. Em suma, a solução descrita provê uma forma de correlacionar evidências e sua origem virtual, permitindo transportar e armazenar tais dados de modo a preservar sua credibilidade. Para isso, a proposta supõe que o sistema sendo monitorado é executado dentro de um contêiner em nuvem. O foco da solução em contêiner se justifica pelo crescimento da adoção de conteinerização nos últimos anos, bem como pela previsão de que este será o modelo de implementação mais usado em aplicações futuras (PIRAGHAJ et al., 2015). A solução tem como alvo específico ataques de injeção de código (CASE et al., 2014), pois estes, quando usados contra uma arquitetura em nuvem, não deixam rastros quando recursos de processamento virtuais são desativados e sua memória é liberada (VöMEL; STüTTGEN, 2013), (CASE et al., 2014). Em particular, têm especial interesse quatro tipos específicos dessa família de ameaças (CASE et al., 2014):

- **Injeção remota de bibliotecas**: Um processo malicioso força o processo alvo a carregar uma biblioteca em seu espaço de memória. Como resultado, o código da biblioteca carregada executa com os mesmos privilégios do executável em que ela foi injetada. Esta estratégia, comumente usada para instalar malwares, pode fazer com que uma biblioteca maliciosa armazenada no sistema seja distribuída por vários processos de uma mesma máquina, dificultando sua remoção (MILLER; TURKULAINEN, 2004).

- **Inline Hooking**: Um processo malicioso escreve código como uma sequência de bytes diretamente no espaço de memória de um processo alvo, e então força este último a executar o código injetado. O código pode ser, por exemplo, um *script* de *shell*.

- **Injeção reflexiva de biblioteca**: Um processo malicioso acessa diretamente a memória do processo alvo, inserindo nela o código de uma biblioteca na forma de uma sequência de bytes, e então força o processo a executar essa biblioteca.

Nessa forma de ataque, a biblioteca maliciosa não existe fisicamente; isso torna tal estratégia de injeção de código potencialmente mais atrativa, pois o carregamento da biblioteca não é registrado no sistema operacional (SO), dificultando a detecção do ataque (FEWER, 2008).

- **Injeção de processo vazio**: Um processo malicioso dispara uma instância de um processo legítimo no estado "suspenso"; a área do executável é então liberada e realocada com código malicioso.

1.3 Justificativa

Uma nuvem computacional é um modelo de infraestrutura no qual recursos compartilhados em quantidade configurável, acessíveis via rede, são alocados e desalocados com esforço mínimo de gerenciamento por parte de um provedor de serviços. (MELL; GRANCE, 2011) Há três modelos principais de comercialização de uso da nuvem (MELL; GRANCE, 2011): *software* como serviço (*Software as a Service* – SaaS), na qual se provê o *software* que será usado pelo cliente; plataforma como serviço (*Platform as a Service* – PaaS), na qual se provê o ambiente para que o cliente desenvolva, teste e execute seu *software*; e, o tipo mais pertinente para este trabalho, Infraestrutura como serviço (*Infrastructure as a Service* – IaaS), na qual são fornecidos recursos computacionais básicos, como processamento e memória, em geral de forma virtualizada.

A virtualização de recursos na nuvem, embora tradicionalmente feita por meio de máquinas virtuais, vêm sendo crescentemente feita também na forma de contêineres. De fato, segundo o "Container Market Adoption Survey 2016", realizado pelas empresas DevOps.com (https://devops.com/) e ClusterHQ (https://clusterhq.com) com 235 empresas que têm desenvolvimento de software como sua atividade fim ou como suporte à atividade fim, 76% dos respondentes utilizam contêineres para melhorar a efi-

ciência do processo de desenvolvimento e em suas arquiteturas de micro-serviços em nuvem. Diferentemente de máquinas virtuais, que envolvem a criação de um *hardware* virtual e de um sistema operacional (SO) acima do sistema nativo e que opera independente deste, a virtualização com contêineres é feita no nível do SO nativo, tem uma implementação mais simples eliminado camadas entre o aplicativo executado e o *hardware* físico. Uma tecnologia bastante utilizada para esse propósito são Contêineres Linux (LXC) (LINUXCONTAINERS.ORG, 2015), que aproveitam-se de funcionalidades como cgroups e namespacing do kernel do Linux para auxiliar no gerenciamento e isolamento de recursos virtuais.

2 ASPECTOS CONCEITUAIS E ECONÔMICOS

2.1 Computação em núvem

Uma nuvem computacional é um modelo de infraestrutura no qual recursos compartilhados em quantidade configurável, acessíveis via rede, são alocados e desalocados com esforço mínimo de gerenciamento por parte de um provedor de serviços. (MELL; GRANCE, 2011) Há três modelos principais de comercialização de uso da nuvem (MELL; GRANCE, 2011): *software* como serviço (*Software as a Service* – SaaS), na qual se provê o *software* que será usado pelo cliente; plataforma como serviço (*Platform as a Service* – PaaS), na qual se provê o ambiente para que o cliente desenvolva, teste e execute seu *software*; e, o tipo mais pertinente para este trabalho, Infraestrutura como serviço (*Infrastructure as a Service* – IaaS), na qual são fornecidos recursos computacionais básicos, como processamento e memória, em geral de forma virtualizada. Uma arquitetura usada hoje nas soluções em núvem são as auto-escaláveis onde recursos são altamente voláteis com recursos sendo alocados e desalocados a qualquer momento. Estas implementações tem a vantagem de usar os recursos de nuvem de uma forma mais eficiênte e menos necessidade de intervenção humana.

2.2 Forense digital e seus desafios atuais

Forense digital (tambem conhecida por forense computacional) é um conjunto de técnicas de coleta e análise de interação entre humanos e computadores de forma que

esta seja aceita em um processo legal. Tal como a forense tradicional, a forense digital se baseia no princípio de Locard. Definido pelo médico francês Edmond Locard, o princípio de Locard estabelece que "Quando um indivíduo entra em contato com outro objeto ou indivíduo, este sempre deixa vestígio deste contato". (RAMOS, 2011) Assim a Forense digital tem por objetivo a investigação de evidências digitais de interações entre homes e máquinas de modo a reconstruir a cadeia de evêntos no passado de forma que suas conclusões cujas conclusões sejam validadas por terceiros. A forense digital quando aplicada a investigação de incidentes em soluções na nuvem enfrenta desafios adicionais relacionados a coleta, transporte e análise da evidência. A primeira delas diz respeito a aceitabilidade de uma evidência em um processo legal. Para que uma evidência seja aceito em um processo legal é necessária que sua cadeia de custódia seja garantida. Cadeia de custódia é o processo de documentação da história cronológica da evidência de modo a saber onde a evidência esteve e quem teve acesso a ela (referencia). O SENASP 2013 diz que: "CADEIA DE CUSTÓDIA: sistemática de procedimentos que visa à preservação do valor probatório da prova pericial caracterizada." Neste ponto as soluções em nuvem com arquiteturas auto-escaláveis possuem uma dificuldade adicional, a volatilidade de seus recursos. Caso uma evidência esteja presente em uma máquina que é desalocada e tem seus recursos liberados esta será pra sempre perdida. A solução seria armazenar a evidência em outro local e seu transporte seguir procedimentos para garantir a cadeia de custódia. Outro desafio é o da reprodutibilidade do processo de coleta. Novamente nas arquiteturas auto-escaláveis em nuvem onde uma VM é desalocada e recursos liberados, reproduzir o processo de coleta não é uma atividade trivial. Um terceido desafio é da preservação da privacidade e respeito a jurisdição. Como nas arquiteturas em nuvem os recursos são complartilhados entre outros usuários, a prática da forense tradicional de remover o recurso físito para posterior análise não pode mais ser usado pois além dos dados relativos a investigação em curso, estes recursos terão também dados de usuários que não estão relacionados a investigação. Por último temos o desafio do volume de informações co-

letadas que hoje sobrecarregam os investigadores forenses, o back-log de investigação hoje é de cerca de 6 meses de dados.

2.3 Conteinerização

A virtualização de recursos na nuvem, embora tradicionalmente feita por meio de máquinas virtuais, vêm sendo crescentemente feita também na forma de contêineres. De fato, segundo o "Container Market Adoption Survey 2016", realizado pelas empresas DevOps.com (https://devops.com/) e ClusterHQ (https://clusterhq.com) com 235 empresas que têm desenvolvimento de software como sua atividade fim ou como suporte à atividade fim, 76% dos respondentes utilizam contêineres para melhorar a eficiência do processo de desenvolvimento e em suas arquiteturas de micro-serviços em nuvem. Diferentemente de máquinas virtuais, que envolvem a criação de um *hardware* virtual e de um sistema operacional (SO) acima do sistema nativo e que opera independente deste, a virtualização com contêineres é feita no nível do SO nativo, tem uma implementação mais simples eliminado camadas entre o aplicativo executado e o *hardware* físico. Uma tecnologia bastante utilizada para esse propósito são Contêineres Linux (LXC) (LINUXCONTAINERS.ORG, 2015), que aproveitam-se de funcionalidades como cgroups e namespacing do kernel do Linux para auxiliar no gerenciamento e isolamento de recursos virtuais.

3 FORENSE DE MEMÓRIA DE MÁQUINAS EM NUVEM

3.1 Credibilidade e aceitabilidade da evidência em processo legal.

O processo de análise forense no evento de um crime digital é descrito no EDIPM - *Enhanced Digital Investigation Process Model* por 4 fases: Identificar, preservar, examinar, apresentar. (calm before the storm) Na fase de preservação da evidência deve ser feita de forma que os autoras descrevem como "forensicamente aceitável" isto é, coletar as evidências de forma que as mesmas sejam aceitas em um processo legal e não sejam invalidadas durante o processo. Segundo (valor probatório do aquivo digital) a aceitabilidade de uma evidência digital em um processo legal deve atender aos seguintes requisitos: Autenticidade Processo pelo qual se pode garantir a autoria do documento eletrônico, ou seja, não permite dúvida quanto à identificação do autor. e Integridade: Permite atestar a "inteireza do documento eletrônico após sua transmissão, bem como apontar eventual alteração irregular de seu conteúdo". Caso haja dúvida sob qualquer um dos requisitos uma perícia técnica pode ser convocada, nesta sera análisada o autor da evidência ou seja sua fonte e se a mesma não foi alterada no processo. Em infra-estrutura física esta coleta era relativamente simples, bastava-se remover o recurso físico, transporta-lo para um laboratório e lá análisar a evidência. A Evidência era mantida em uma sala-cofre onde o acesso era controlado. A reprodutibilidade do processo de coleta e a manutenção da integridade da evidência eram tarefas bem diretas. A computação em núvem, especialmente as de infraestrutura auto-

escalável troxeram um conjunto de desafios para se atingir este requisito. O recurso não pode mais ser removido pois o mesmo é utilizado por outros usuários não relacionados a investigação, fazê-lo constituiria violação de privacidade. A volatilidade dos recursos tornou a verificação do seu autor um processo mais complexo pois o recurso que a gerou pode não existir mais. A integridade da evidência também tornou-se mais complexa pois ela precisa ser coletada, transportada e armazenada. O processo de cadeia de custôdia ganhou grande visibilidade neste quesito. Violação de qualquer uma das caracteristicas citadas anteriormente põe em dúvida a credibilidade da evidência.

3.2 Volume de dados para coleta

O processo de coleta da evidência na forense digital herdou suas práticas da forense tradicional onde isola-se cena do crime e coletam-se as evidências presentes. Transportando para a forense digital criou-se o hábito de se realizar cópia bit a bit da informação que se deseja investigar. No passado, com as soluções tendo bem menos capacidade de memória, disco e tráfego, tal prática não trazia problemas. Nas atuais soluções, aplicações e arquiteturas em núvem o volume de dados aumentou muito. Em (encontrar a data) investigadores forenses tinham em média 6 meses de backlog para analisar. Em conversas informais com analistas forenses é comum a métrica de em média apenas 2% do material coletado ser útil a análise. Encontrar uma forma de armazenar menos informações de modo a tornar a fase de análise mais rápida e eficiênte ajudará nas investigações.

3.3 Privacidade e jurisdição

Na metodologia tradicional de coleta de evidências para análise isola-se o ambiente e as evidencias são removidas. Transportado para a forense digital temos a prática de remover o equipamento para realização de cópia bit a bit da evidência. Nas soluções

de infra estrutura física esta prática não trás problemas, os objeto ou indivíduos sob investigação estão diretamente relacionados ao equipamento removido. Nas soluções em nuvem esta prática não pode mais ser utilizada pois como o recurso físico é compartilhado por vários usuários não envolvidos na investigação, remove-los configura violação de privacidade. Um complicador a mais é o fato de os dados não estarem armazenados no mesmo território em que a investigação é realizada demandando acordos de cooperação jurídica entre as partes o que nem sempre é possível. Neste cenário encontrar uma forma de coletar a evidência sem violar jurisdição e privacidade são de grande importância num futuro próximo.

4 REVISÃO BIBLIOGRÁFICA

Existem vários aspectos relativos à análise forense na nuvem, indo desde a coleta de informações até a garantia da cadeia de custódia de evidências. Para uma discussão mais estruturada dos trabalhos disponíveis na literatura sobre o tema, a seguir eles são apresentados com base nos diferentes aspectos que abordam.

4.1 Acessar e coletar as informações de memória das máquinas virtuais em nuvem

Diversos trabalhos de análise forense na nuvem se concentram na coleta de dados "após o fato", ou seja, após a intrusão ser detectada (REICHERT; RICHARDS; YOSHIGOE, 2015; POISEL; MALZER; TJOA, 2013; DYKSTRA; SHERMAN, 2013; GEORGE; VENTER; THOMAS, 2012; SANG, 2013). Os processos de coleta descritos nesses trabalhos podem ser iniciados de forma manual ou automaticamente, via integração com um mecanismo de detecção de intrusão. No caso específico de memória volátil, tal forma de coleta não consegue descrever como era a memória antes da intrusão, pois o processo só é acionado depois da detecção do ataque. Tal limitação pode trazer prejuízos à investigação, dado que algumas análises dependem exatamente da capacidade de se comparar dois momentos da memória (CASE et al., 2014). Entre os trabalhos estudados, a única proposta encontrada que leva tal necessidade em consideração é (DEZFOULI et al., 2012), que propõe que o dado seja armazenado no próprio equipamento sob análise. Infelizmente, entretanto, a aplicação de tal abor-

dagem no cenário em nuvem é pouco viável, pois pode levar à perda de informações importantes caso a máquina virtual ou contêiner seja desativada, tendo seus recursos liberados.

Existem ainda trabalhos voltados à coleta de informações durante a execução do sistema, nos quais os dados são constantemente coletados sem distinção do que aconteceu antes ou depois do fato de interesse. Esse é o caso de trabalhos como (POISEL; MALZER; TJOA, 2013; DYKSTRA; SHERMAN, 2013; SANG, 2013), que adotam a estratégia de isolar e parar a máquina virtual para em seguida realizar o processo de coleta. Embora interessantes, as abordagens descritas nesses trabalhos podem levar a um elevado volume de dados coletados, além de também não tratarem o cenário em que é necessário coletar evidências quando os recursos virtuais contendo tais informações são liberados.

4.2 Capacidade de reproduzir o processo e obter os mesmos resultados

Se, durante uma análise forense, analistas diferentes obtêm resultados distintos ao executar o mesmo procedimento de coleta, a evidência gerada não tem credibilidade, inviabilizando seu uso em um processo legal. Por essa razão, a reprodutibilidade do processo de coleta é uma parte importante da geração de evidências para análise forense. Infelizmente, entretanto, nenhuma das propostas encontradas na literatura atualmente permite tal reprodutibilidade em cenários de nuvem em que máquinas virtuais ou contêineres são desativados e seus recursos físicos liberados: todas elas dependem da existência do recurso virtual para a repetição do processo de coleta.

4.3 Não violar privacidade ou jurisdição das partes não envolvidas na investigação

Em um ambiente de nuvem pública, remover o *hardware* para análise posterior pode levar à violação de privacidade de usuários, uma vez que o multi-inquilinato desse cenário faz com que uma mesma máquina física guarde informações de diversos clientes, alguns dos quais podem não estar envolvidos na investigação em curso. Diversos trabalhos na literatura tratam esse problema adequadamente, por meio das duas estratégias principais: a primeira, adotada em (REICHERT; RICHARDS; YOSHIGOE, 2015; GEORGE; VENTER; THOMAS, 2012; POISEL; MALZER; TJOA, 2013; DYKSTRA; SHERMAN, 2013), consiste em coletar dados pertinentes à investigação e armazená-los fora da nuvem; a segunda, empregada em (SANG, 2013) e que constitui um caso específico de (GEORGE; VENTER; THOMAS, 2012), depende da cooperação do provedor de serviços de nuvem para conseguir as informações necessárias à investigação. Depender do provedor de serviços de nuvem é uma estratégia pouco recomendada, entretanto, pois (1) o volume de dados de usuários pode forçar os provedores a limitar o tamanho dos *logs* armazenados, e (2) caso ocorra uma indisponibilidade causada por um ataque, o objetivo do provedor será o de restabelecer o serviço, não necessariamente o de preservar evidências(ALQAHTANY et al., 2015).

4.4 Garantir a cadeia de custódia da evidência

Dentre os trabalhos analisados, apenas (SANG, 2013) aborda a questão da garantia da cadeia de custódia. Especificamente, o trabalho emprega *hashes* para verificar a integridade da evidência, permitindo a detecção de alterações na mesma, embora não explique os mecanismos que poderiam ser utilizados para impedir acesso não autorizado (e, assim, potencial alteração) aos próprios hashes. As propostas dos outros autores concentram-se apenas no aspecto técnico da coleta, sem discutir claramente

garantia de custódia mas apenas mencionando que as evidências devem ser coletadas de forma “forensicamente aceitável”.

4.5 Resumo

A Tabela 1 mostra um comparativo das soluções estudadas, considerando os aspectos discutidos nesta seção, posicionando as contribuições da proposta apresentada neste trabalho.

Table 1: Comparativo de soluções de coleta de informações de memória de máquinas em nuvem para análise forense

	Dizang (esta proposta)	(GEORGE; VENTER; THOMAS, 2012)	(POISEL; MALZER; TJOA, 2013)	(DYKSTRA; SHERMAN, 2013)	(BARBARA, 2014)	(REICHERT; RICHARDS; YOSHIGOE, 2015)	(SANG, 2013)	(DOLAN-GAVITT et al., 2011)	(ALJAEDI et al., 2011)	(DEZFOULI et al., 2012)	(BAAR; BEEK; EIJK, 2014)
Coleta é contínua?	✓	✘	✘	✘	✘	✘	✓	✘	✘	✓	✓
Reproduz o processo sem a VM?	✓	✘	✘	✘	✘	✘	✘	✘	✘	✘	✘
Garante cadeia de custódia?	✓	✘	✘	✘	✘	✓	✓	✘	✘	✘	✓
Preserva jurisdição e privacidade?	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

5 PROPOSTA DE PROJETO

5.1 Métodos de pesquisa

A primeira parte da pesquisa será realizada implementando a solução de coleta e executando-a em uma máquina virtual em um notebook (descrever notebook) de modo a poder provar que é possível relacionar a evidência coletada a sua origem mesmo se esta não existir mais A segunda parte da pesquisa envolve o transporte via conexão segura para uma máquina física fora da virtual via conexão segura. A terceira parte é a de se realizar uma análise nas evidências coletadas.

5.2 O que foi feito até então

Até então foi realizado a relação entre a evidência coletada e sua origem via hash de identificação do contêiner. Destruimos o contêiner, recriamos o contêiner e a evidência coletada tem o mesmo hash.

5.3 Limitações

Como a solução descrita tem como foco coletar informações de memória no espaço do usuário (*user space*), ela não consegue acessar o espaço de kernel (*kernel space*). Assim, Dizangem princípio não provê suporte a técnicas de investigação de malware que se baseiam em informações do *kernel space*, como, por exemplo, a comparação de informações do bloco do ambiente do processo (*Process Environment Block – PEB*),

que ficam no *user space*, com informações do descritor de endereços de memória virtual (*Virtual Address Descriptor – VAD*), que fica no *kernel space*. Análise de ameaças que realizam manipulação direta dos objetos do kernel (D.K.O.M.– *Direct Kernel Object Manipulation*) também não se beneficiam com a solução aqui proposta.

5.4 Contribuições

A contribuição é o de contribuir para realização de análises e consequentemente utilizar as descobertas em processos legais.

REFERENCES

ALJAEDI, A. et al. Comparative analysis of volatile memory forensics: Live response vs. memory imaging. In *IEEE 3rd Int. Conf. on Privacy, Security, Risk and Trust*. [S.l.: s.n.], 2011. p. 1253–1258.

ALQAHTANY, S. et al. Cloud forensics: A review of challenges, solutions and open problems. In *Int. Conference on Cloud Computing (ICCC)*. [S.l.]: IEEE, 2015. p. 1–9.

BAAR, R. B. van; BEEK, H. M. A. van; EIJK, E. J. van. Digital Forensics as a Service: A game changer. *Digital Investigation*, Elsevier Ltd, vol. 11, p. S54–S62, 2014. ISSN 17422876. Available from Internet: <dx.doi.org/10.1016/j.diin.2014.03.007>.

BARBARA, D. *Desafios da perícia forense em um ambiente de computação nas nuvens*. [S.l.], 2014. <revista.uniplac.net/ojs/index.php/tc_si/article/view/1911>.

CASE, A. et al. *The Art of Memory Forensics: Detecting malware and threats in Windows, Linux and Mac memory*. [S.l.]: Wiley, 2014.

DEZFOULI, F. et al. Volatile memory acquisition using backup for forensic investigation. In *Int. Conf. on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec)*. [S.l.: s.n.], 2012. p. 186–189.

DOLAN-GAVITT, B. et al. Virtuoso: Narrowing the semantic gap in virtual machine introspection. In *IEEE Symposium on Security and Privacy*. [S.l.: s.n.], 2011. p. 297–312. ISSN 1081-6011.

DYKSTRA, J.; SHERMAN, A. Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques. *Digital Investigation*, Elsevier Ltd, vol. 9, p. S90–S98, 2012. ISSN 17422876. (Proc. of the 12th Annual DFRWS Conference). Available from Internet: <dx.doi.org/10.1016/j.diin.2012.05.001>.

DYKSTRA, J.; SHERMAN, A. T. Design and implementation of FROST: Digital forensic tools for the OpenStack cloud computing platform. *Digital Investigation*, Elsevier Ltd, vol. 10, p. S87–S95, 2013. ISSN 17422876. (Proc. of 13th Annual DFRWS Conference). Available from Internet: <dx.doi.org/10.1016/j.diin.2013.06.010>.

FEWER, B. S. Reflective DLL Injection. no. October, 2008.

GEORGE, S.; VENTER, H.; THOMAS, F. Digital Forensic Framework for a Cloud Environment. In *IST Africa*. Tanzania: IIMC, 2012. p. 1–8. ISBN 9781905824342.

GILBERT, P.; SUJEET, S. *Advances in Digital Forensics IV*. 1. ed. Orlando: Springer-US, 2008. vol. 1. ISSN 1098-6596. ISBN 9788578110796.

LINUXCONTAINERS.ORG. *Linux Containers (LXC)*. 2015. Available from Internet: <https://linuxcontainers.org/lxc/introduction/>.

MELL, P.; GRANCE, T. *The NIST definition of cloud computing*. 2011. 7 p. NIST SP 800-145. <csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>. Available from Internet: <www.mendeley.com/research/the-nist-definition-about-cloud-computing/>.

MILLER, M.; TURKULAINEN, J. *Remote Library Injection*. 2004. Tech. Report: <www.nologin.org/Downloads/Papers/remote-library-injection.pdf>.

MORSY, A. M.; GRUNDY, J.; MULLER, I. An Analysis of the Cloud Computing Security Problem. In *APSEC Cloud Workshop*. Sydney, Australia: [s.n.], 2010. Available from Internet: <https://arxiv.org/abs/1609.01107>.

PIRAGHAJ, S. et al. A framework and algorithm for energy efficient container consolidation in cloud data centers. In *IEEE Int. Conf. on Data Science and Data Intensive Systems (DSDIS)*. [S.l.: s.n.], 2015. p. 368–375.

POISEL, R.; MALZER, E.; TJOA, S. Evidence and cloud computing: The virtual machine introspection approach. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 4, no. 1, p. 135–152, 2013. ISSN 20935374 (ISSN). Available from Internet: <citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.469.937>.

RAHMAN, S.; KHAN, M. N. A. Review of live forensic analysis techniques. *International Journal of Hybrid Information Technology*, vol. 8, no. 2, p. 379–388, 2015. Available from Internet: <www.sersc.org/journals/IJHIT/>.

RAMOS, M. G. *Do Valor Probatório do Arquivo Digital*. PhD Thesis (PhD) — Universidade de Brasilia, 2011.

REICHERT, Z.; RICHARDS, K.; YOSHIGOE, K. Automated forensic data acquisition in the cloud. *IEEE Int. Conf. on Mobile Ad Hoc and Sensor Systems*, p. 725–730, 2015.

SANG, T. A log-based approach to make digital forensics easier on cloud computing. *Intelligent System Design and Engineering Applications (ISDEA)*, p. 91–94, 2013.

VöMEL, S.; STüTTGEN, J. An evaluation platform for forensic memory acquisition software. *Digit. Investig.*, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, vol. 10, p. S30–S40, 2013. ISSN 1742-2876. Elsevier Science Publishers. Available from Internet: <http://dx.doi.org/10.1016/j.diin.2013.06.004>.