Coletando dados de memória de uma máquina em nuvem para análise forense

Hamilton Fonte II
Universidade de São Paulo (USP)
Escola Politécnica - Engenharia de Computação
Programa de Pós Graduação em Engenharia Elétrica
São Paulo, SP, Brasil
Email: hamiltonii@gmail.com

Marcus Simplício Jr.
Orientador
Universidade de São Paulo (USP)
Escola Politécnica - Engenharia de Computação
Programa de Pós Graduação em Engenharia Elétrica
São Paulo, SP, Brasil

***Abstract*—Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim. Donec pede justo, fringilla vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut, imperdiet a, venenatis vitae, justo. Nullam dictum felis eu pede mollis pretium.**

I. Introdução

Aumento do uso de soluções de virtualização e a implementação de arquiteturas em nuvem que escalam automáticamente [1] trouxe a questão da volatilidade das máquinas virtuais. Uma aplicação hospedada na nuvem sob um pico de uso pode clonar máquinas e adiciona-las ao grupo para atender a demanda. Passado este pico, as máquinas que foram clonadas são despejadas, seus recursos liberados e o conjunto retorna ao tamanho inicial. Com as ameaças que atuam diretamente na memória sem deixar rastros no disco da máquina afetada, se essas máquinas forem usadas para algum evento ilícito, as evidências do acontecimento contidas nelas serão para sempre perdidas.

Do ponto de vista forense, praticantes e pesquisadores concordam que aspectos de multi-inquilino e multi-jurisdição próprios soluções em nuvem figuram entre as principais dificuldades para coleta de evidência [2]. O aspecto multi-inquilino impede a remoção do hardware pois como ele é compartilhado com vários usuários, removê-los seria uma violação de privacidade de usuários não relacionados a investigação. Por fim a característica distribuída pode alocar informação relevante a investigação em vários países dificultando por razões jurídicas a obtenção da mesma [3].

Este documento esta organizado da seguinte forma: Na seção II falamos brevemente sobre soluções em nuvem, na seção III falamos sobre a evolução da forense e os desafios que as soluções em nuvem trouxeram para a forense, na seção IV analisamos os trabalhos na área de forense de memória, na seção V descrevemos a solução proposta para resolver os problemas descritos em III, na VI expomos nossas conclusões e na VII elencamos os trabalhos futuros.

II. Adoção de arquiteturas em nuvem

A nuvem é um sistema em que recursos computacionais são oferecidos como serviço onde os usuários são cobrados pelo seu uso. A infra-estrutura é composta de máquinas fisicas contendo cada uma um número variável de máquinas virtuais que implementam este serviço [18]. Há três modelos de comercialização de uso da nuvem, plataforma como serviço (PAAS), software como serviço(SAAS) e pertinente e este trabalho Infraestrutura como serviço (IAAS). O uso de soluções baseadas em nuvem tem crescido muito ultimamente. Por exemplo, de janeiro de 2016 a maio de 2016 mais de 1000 bases de dados foram migradas para a AWS. [1].

Uma forma mais recente de arquitetura em nuvem, introduzido em 2008, os Containers Linux (LXC) proveram uma série de ferramentas para tirar vantagens das funcionalidades de cgroups e namespacing do kernel do Linux. Este conceito foi evoluindo e várias soluções de containerização sugiram como IMCTFY, Rocket e Docker. Container é uma forma de isolamento entre processos onde os containers partilham o mesmo kernel e tem sido usado para desenvolver serviços baseado em vitualização. A adoção de container tem crescido muito, segundo o ”Container Market Adoption Survey 2016”, das 235 empresas que responderam o survey 76% delas utilizam containers em ambiente de produção.

III. Forense de memória em nuvem

A evolução da forense digital pode ser descrita em 3 fases [19] A primeira ad-hoc se caracteriza pela falta de estrutura, processos, ferramental e objetivos. Nesta fase evidências apresentadas em processos legais eram descartadas com base em erros procedurais e falta de garantias de acurácia ou cadeia de custódia. A segunda fase começa a estruturação da forense, nesta surgem as primeiras políticas e processos de coleta, armazenamento, transpórte e análise da evidência. O primeiro processo proposto por Palmer G. em 2001 na primeira Digital Forensics Research Conference ficou conhecido como DIP, o próprio Palmer julgava o modelo incompleto. Em 2002 Reith M. propôs o Abstract Digital Forensics Model que adicionava ao DIP estágios que faltava a este. Em 2003 Carrier e Spafford propuseram o Integrated Digital Investigation Process. Baseado nas técnicas e teorias da forense física e finalmente

em 2004 Baryamureeba e Tushabe propuseram o Enhanced Digital Investigation Process Model, uma evolução de Carrier e Spafford. Importante notar que esses modelos foram todos propostos antes que a forma atual de computação em nuvem estivesse disponível as massas. [20]

Nesta fase surgiram as ferramentas que tinham por objetivo principal coletar evidências de forma que fossem legalmente aceitáveis, para isso alguns requisitos precisam ser atendidos:

1) O processo de coleta precisa ser repetível.
2) O processo de coleta precisa ser confiável.
3) O processo de coleta precisa preservar a evidência.

A terceira fase se caracteriza pela migração do ferramental de soluções pontuais para soluções empresariais. Conceitos como coleta em tempo real e Forense como serviço emergem nesta fase.

A utilização crescente de virtualização, ferramentas online e hospedagem em nuvem [1], está criando dificuldades para a coleta de informações, análise e utilização em processos legais [21]. A funcionalidade de elasticidade de carga ofertada pelos provedores de nuvem por meio da qual infraestrutura pode ser alocada e desalocada dinamicamente, trouxe o problema da volatilidade dos dados nas máquinas virtuais. Com algumas ameaças que não deixam evidências em disco [22], a memória de uma máquina virtual despejada de um pool e seus recursos liberados seria para sempre perdida e com ela evidências importantes. O simples armazenamento do conteúdo da memória não satisfaz o requisito jurídico de se repetir o processo e conseguir os mesmos resultados. A abordagem de armazenar constantemente todas as alterações da memória não contribui para a solução do crescente backlog de dados que os investigadores tem para analizar [23].

O ferramental forense disponível hoje está pouco adaptado a desafios trazidos pela nuvem [3], focam em completude e poucos geram evidências aceitáveis em um processo jurídico [7]. A cadeia de custódia, um processo de coleta e armazenamento de evidências que visa garantir que a evidência não foi alterada, destruída ou manipulada por pessoas não autorizadas, é pouco abordada nas soluções existentes hoje.

IV. Trabalhos relacionados

- **Digital forensics framework for a cloud environment [4]** : Arcabouço para coleta de dados de máquinas virtuais. Possui duas formas de acionamento, a manual e a automática, integrada com algum sistema de detecção de ameaça. Quando acionado, escuta a rede, determina qual máquina é objeto de investigação, coleta informações de log e tráfego de rede e associa ao usuário da respectiva máquina. Propõe o armazenamento das evidências em local fora da núvem para escapar de problemas de jurisdição e multi-inquilino mas tem inteligência para usar a própria nuvem como armazenamento caso o espaço fora acabe.
  A proposta dá a entender que é aplicável apenas a um sistema virtual estático, onde o número e organização das máquinas é constante. De informação volátil coleta apenas tráfego de rede, não coleta memória. Com a forma de acionamento descrito ele não consegue descrever, com as evidências, como era o sistema antes do ataque. Apesar de armazenar a evidência fora da nuvem, não da detalhes de como garante que a evidência não foi alterada ou destruída no transporte até o local de armazenamento nem como controla o acesso a evidência.
  Quando comparado a este trabalho, a presente proposta tem por vantagens a utilização de container para associar a evidência a sua origem tornando o processo independente de máquina e permitindo que seja repetido mesmo se a máquina de onde se originaram os dados não existir mais. Com a implementação de uma janela de x dias de coleta antes da detecção do ataque é possível descrever, através de evidência, como era o sistema antes do mesmo. Com isso a solução apresentada consegue evidências em um cenário de infra-estrutura dinâmica. São tomadas precauções para garantir que os dados não foram alterados ou destruídos no transporte via TLS os para um local fora da nuvem e o acesso a mesma é controlado.

- **Evidence and cloud computing the virtual machine introspection approach [5]** : Descreve um método de coleta de informações de máquinas em nuvem através da técnica de introspecção em máquina virtual, onde se acessa os dados das máquinas virtuais através do hypervisor. Propõe que o processo seja disparado automaticamente integrado a um sistema de detecção de ameaça mas também suporta acionamento manual.
  A técnica descrita cobre apenas o processo de coleta de informações, não explica onde ou como elas serão armazenadas. No que tange as informações de memória, como os endereços de memória são os do host, estes precisam ser traduzidos para que a análise forense seja feita. Segundo a comunidade, tal estratégia é imune a técnicas anti- forenses empregadas por usuários maliciosos pois está localizada fora da máquina virtual. Como a abordagem não tem conhecimento do que está rodando dentro da máquina precisa de uma copia bit a bit da evidência. Embora pareça possível, não descreve como lida com o cenário onde uma máquina é despejada do pool e os recursos liberados.
  Quando comparado a este trabalho, a presente proposta tem por vantagens ser um arcabouço para coleta e armazenamento de evidências. Usa-se uma estratégia diferente pois coleta-se a memória diretamente de dentro da máquina virtual onde se evita o problema do gap semântico próprio das soluções por introspecção. Como não precisa realizar tradução de endereços, a presente proposta consegue realizar uma coleta onde os dados já são úteis para análise e pode direcionar a mesma pois tem o conhecimento do que está rodando na máquina. De acordo com a comunidade é mais sucetível a técnicas anti-forenses.

- **Design and implementation of FROST: FoRensic tools**

for Open STack [6] : Arcabouço para coleta de dados de máquinas virtuais através da API do hypervisor. Isola a máquina virtual afetada do pool original para realização da coleta. Precisa ser acionado quando uma ameaça é detectada. É o mais bem acabado arcabouço de todas as propostas encontradas até agora mas ao detalhar o processo de armazenamento não explica como garante que a evidência não será destruida ou alterada no transporte até o armazenamento nem como controla o acesso a evidência. Por estar integrado ao Open Stack o arcabouço depende de cooperação do provedor de serviços de nuvem onde ele está rodando, isso é considerado problemático pela comunidade pois a prioridade do mesmo é manter o serviço funcionando e não coletar evidencias forenses. Como está na mesma camada do hypervisor não conhece o que está rodando dentro da máquina. Depende da existência da máquina virtual para realização da coleta.
Quando comparado a este trabalho, a presente proposta tem por vantagens a utilização de container para associar a evidência a sua origem tornando o processo independente de máquina e permitindo que seja repetido mesmo se a máquina de onde originaram os dados não existir mais. Coma implementação de uma janela de x dias de coleta antes da detecção do ataque é possível descrever, através da evidência, como era o sistema antes do mesmo. Não depende de cooperação do provedor do serviço de nuvem. A presente proposta também consegue realizar uma coleta onde os dados já são úteis para análise e pode direcionar a mesma pois tem o conhecimento do que está rodando na máquina.

- **Automated Forensic Data Acquisition in the Cloud [7]** : Propõe um modelo que tira instantâneos de máquinas virtuais atrelado a algum mecanismo de detecção de ameaça baseado no hypervisor. Usa o Google Rapid Response para salvar as informações coletadas fora da núvem de forma a driblar os problemas de multi-jurisdição e multi-inquilino. Descreve satisfatóriamente como evita que a evidência seja alterada ou destruída no transporte até o armazenamento e como controla o acesso a evidência.
  O modelo proposto só começa a coletar evidência após a detecção da ameaça e toma um instantâneo da máquina toda o que já foi julgado pela comunidade como um processo custoso em termos de espaco em disco e piora o problema do volume de dados a ser analisado. Pessoalmente acho arriscado depender de instantâneos pois caso precise, repetir o processo de coleta pode não ser possível. Um exemplo é editar um disco virtual que estava atrelado a uma máquina virtual da qual se gerou os instantâneos, tal ação pode levar a perda de dados.
  Como métrica, o autor relaciona o tamanho da memória alocada na máquina virtual com o tempo necessário para gerar o instantâneo de acordo com a tabela 1 abaixo
  Quando comparado a este trabalho, a presente proposta tem por vantagens coletar apenas a informações de memória e usar a janela de coleta de **x** dias antes do ataque para manter sob controle a quantidade de informação que precisa ser analisada. Tomando como referência a tabela acima, conseguiremos um menor tempo de coleta da informação de memória pertinente a investigação, permitindo um menor espaço de tempo entre as coletas, gerando menos impacto na performance da aplicação e mais dados para a investigação. Propondo a autilização de container para associar a evidência a sua origem, tornamos o processo independente de máquina.

TABLE I
MEMÓRIA ALOCADA X TEMPO DE CAPTAÇÃO

Memória alocada na VM	Tempo geração snapshot
512 Mb	15 segundos
1 Gb	20 segundos
4 Gb	36 segundos

- **A log based approach to make digital forensics easier on cloud computing [8]** : Método sugere salvar a informação coletada fora da núvem de modo a driblar os problemas de multi-inquilinato e multi-jurisdição, usa um mecanismo de hash para garantir a autenticidade e integridade da informação mas não dá detalhes da implementação e não descreve como controla o acesso a evidência armazenada. Segundo o próprio autor, o método não funciona em IaaS. Precisa da cooperação do provedor de nuvem pois depende das informações que este último decidiu adicionar ao log. O método não é aplicável a coleta de informações de memória.
  A proposta não coleta dados de memória por decisão do autor, esta proposta entrou na lista pela abordagem baseada em log. Neste quesito, a presente proposta é a melhor pois garante que a evidência não foi alterada ou destruida no transporte e o acesso a mesma é controlado. No âmbito da informação coletada, a presente proposta não depende das decisões do provedor de nuvem sobre o que guardar no log para conseguir a evidência.

- **Volatile memory acquisition using backup for forensic investigation [9]** : Técnica desenvolvida para dispositivos móveis que sugere a utilização do próprio como repositório das evidências coletadas da memória. Para manter a utilização de espaço ao mínimo sugere manter apenas o último estado conhecido da memória.
  É uma técnica interessante do ponto de vista de estratégia de armazenamento quando guarda apenas o último estado da memória. Essa abordagem porém perde a informação do momento do ataque e não consegue descrever o sistema antes do mesmo. Do resto da proposta não é aplicável para este projeto pois, armazenando a evidência na máquina a mesma seria perdida quando a máquina fosse despejada do pool e seus recursos liberados. A cadeia de custódia não é abordada na proposta.

TABLE II
TAMANHO DO TRACE COLETADO DE VÁRIOS PROGRAMAS

Sistema Operacional	Programa	Tamanho do Trace
Windows	getpid	3549
	gettime	7715
	pslist	302082
	lsmod	195488
	getpsfile	49588
	getdrvfile	194765
	getpid	133047
Linux	gettime	75074
	pslist	6107214
	lsmod	1936439
	getpsfile	14752561
	getpid	18242
Haiku	gettime	9982
	pslist	362127
	lsmod	850363
	getpsfile	249663
	getdrvfile	522299

- **Narrowing the semantic gap in virtual machine introspection [10]** : Esta proposta é uma combinação da técnica de introspecção em máquina virtual e a integração com a API do hypervisor. O principal objetivo é diminuir o gap semântico para facilitar a análise da evidência. Para isso o autor implementa um API para transformar dados de baixo nível em informação de alto nível. Depende de cooperação do provedor de serviço de nuvem, não tem conhecimento da máquina hospedada e não vai além da coleta, não descreve como resolve a cadeia de custódia. Tem a vantagem de ser imune a técnicas anti-forenses.
  Como métrica o autor relaciona o tamanho médio do trace gerado a partir da uma evidência de memória coletada de alguns processos em vários sistemas operacionais de acordo com a tabela 2 abaixo. Essa métrica pode se relacionar a presente proposta como o volume de informação extraida de uma evidência.
  Quando comparado a este trabalho, a presente proposta tem por vantagens ser arcabouço de coleta e armazenamento de evidências não apenas um método de coleta. Empregando a estratégia de realizar a coleta diretamente na máquina não tem o problema do gap semântico próprio das soluções baseadas em introspecção. Como conhece o contexto do que está rodando dentro da máquina virtual podemos direcionar a coleta de modo que seja mais eficiênte. O autor usa outras métricas voltadas ao processamento da evidência antes de sua análise para diminuir a quantidade de informação a se analisar mas não dá detalhes de como esse processamento.

- **Digital Forensics as a Service: A game changer [11]** : Esta proposta é focada em uma mudança no armazenamento e forma de trabalho dos peritos forenses. Propõe que a forense seja oferecida como um serviço e que as evidências sejam armazenadas em um local centralizado com o devido controle de acesso e garantia de integridade da evidência. Descreve a arquitetura de armazenamento, qual o perfil que deve ter acesso a evidência e como é este acesso. Esta proposta não é focada apenas em incidentes na nuvem mas em qualquer outro incidente.
  Embora seja uma ótima proposta de armazenamento de evidências e controle de acesso a elas, ele não descreve o processo de coleta nem de transporte. É uma proposta focada mais na solução de problemas relacionados a manipulação da informação após a coleta, transporte e armazenamento. Não toca no assunto de coleta qualquer que seja, em nuvem ou fisica.

- **Live Digital Forensics in a Virtual Machine [12]** : Proposta para coletar informações de memória de máquinas virtuais através de instantâneos das mesmas. O metodo de coleta envolve tomar o instantâneo da máquina, no diretório onde o mesmo foi armazenado pegar o arquivo referente a memória e abri-lo / analisá-lo usando um programa de leitura de memória de mercado. O autor não trás detalhes do transporte, armazenamento ou controle de acesso. Precisa que a máquina exista para conseguir coletar a informação e o processo é dependente de intervenção humana. Analisando com mais cuidado é possível repetir a coleta mesmo sem a máquina existir uma vez que temos o instantâneo mas o autor não dá detalhes do caso.
  Quando comparado a este trabalho, a presente proposta tem por vantagens a menor quantidade de informação necessária à investigação através da implementação da janela de x dias antes do incidente. Como o processo é automático, uma vez disparado não requer intervenção humana. A presente proposta descreve como garante a cadeia de custódia da evidência e consigue reproduzir o processo de coleta mesmo se a máquina não existir mais pois a evidência está atrelada ao container.

- **Comparative Analisys of volatile memory forensics [13]** : Levantamento sobre o impacto da realização de forense de memória ao vivo nas máquinas virtuais na nuvem. Mostra que a quantidade de informações referentes a processos não alocados e página de memória perdida é um ponto a considerar quando se inicia uma ferramente para análise de memória em uma máquina já funcionando como.
  Como métrica o autor relaciona a porcentagem média de processos que permanecem na memória antes e depois que a ferramenta de coleta foi disparada com a quantidade de memória alocada na VM de acordo com a tabela 3 abaixo.

TABLE III
MÉDIA DE PROCESSOS QUE PERMANECERAM NA MEMÓRIA

Memória alocada a VM	% antes ativação	% depois ativação
1 Gb	78,33	61,66
512 Mb	73,33	46,66
256 Mb	50,55	35,00

Em outra métrica o autor relaciona a porcentagem média de páginas de memória alterada nos métodos de **% análise ao vivo**, onde a coleta é realizada com o sistema rodando e **% cópia de memória** onde a máquina virtual é pausada para a realização de coleta de informação com a quantidade de memória alocada de acordo com a tabela abaixo. Em ambos os casos as duas métricas são comparações entre o estado da memória antes e depois da inicialização da ferramenta de coleta.

TABLE IV
MÉDIA DE PÁGINA DE MEMÓRIA ALTERADA

Memória alocada a VM	% análise ao vivo	% cópia de memória
1 Gb	7,99	5,95
512 Mb	32,53	8,75
256 Mb	52,37	25,46

Quando comparado a este trabalho a presente proposta tem por vantagem estar sempre em execução, sujeitando-se aos efeitos de escalonamento do processo e gerenciamento de páginas de memória pelo sistema operacional, assim não gera as perdas de informação de processos não alocados e páginas de memória referentes ao rearranjo que o kernel faz quando uma nova aplicação é iniciada.

Tabela comparativa das soluções

TABLE V
COMPARATIVO DE SOLUÇÕES

	Coleta é continua?	Precisa de tradução de endereços para análise?	É independente de VM?	Conhece o que esta rodando na VM?	Garante cadeia de custódia?	Preserva evidência de memória volátil?
[4]	✘	✘	✘	✓	✘	✘
[5]	✘	✓	✘	✘	✘	✓
[6]	✘	✘	✘	✘	✘	✘
[14]	✘	✘	✘	✘	✘	✘
[7]	✘	✘	✘	✓	✓	✘
[8]	✓	✘	✓	✓	✘	✘
[10]	✘	✘	✘	✘	✘	✓
[13]	✘	✘	✘	✓	✘	✓
[9]	✓	✘	✘	✓	✘	✓
[11]	✓	✘	✘	✘	✓	✘

V. SOLUÇÃO PROPOSTA

A. Objetivos

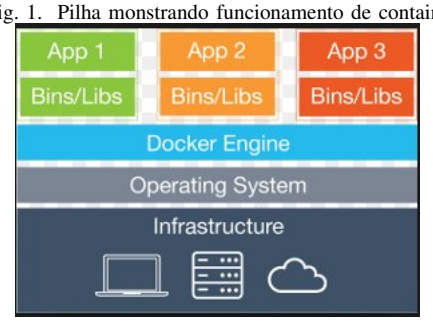
O presente proposta tem os seguintes objetivos:

- Coletar memória de uma máquina virtual de modo a conseguir identificar os 4 tipos de ataque listados anteriormente.
- Coletar memória de uma máquina virtual de modo a conseguir identificar sua fonte mesmo se a máquina virtual não existir mais.
- Coletar memória suficiente para conseguir descrever o sistema antes e depois do incidente.
- Armazenar a memória coletada de modo a garantir sua integridade, confidencialidade, não violar jurisdição e não violar privacidade de outros usuários no host.

B. Descrição

Nas soluções com infra-estrutura física a máquina é persistente. Associar uma copia da memória, a imagem de um disco ou pacotes trafegando na rede a uma máquina é tarefa simples. Com as soluções de infra virtual, em especial as auto-escaláveis, a máquina deixou de ser persistente e tornou-se volátil. Para resolver o problema da identificação da fonte precisamos encontrar outra forma persistente para identificar a fonte da evidência coletada. Para isto usamos containeres. Embora o container seja uma peça de software e por consequência também é volátil, a imagem compilada e sua execução na forma de container estão atrelados a um hash que os identificam, a pilha de um container pode ser visto na Figura 1.

Fig. 1. Pilha monstrando funcionamento de container



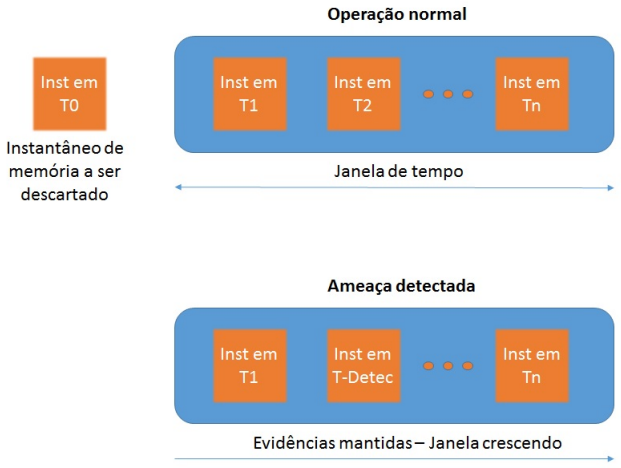
A solução proposta por este trabalho, para resolver o problema de associação da evidência a sua origem de modo que o processo seja reprodutível, pausa a execução do container e coleta um instantâneo da memória dos processos sob sua execução. Este processo é executado em intervalos de tempo conhecidos de modo a se ter uma evolução da história da memória dos processos. Em um sistema derivado do linux (Ubuntu 14.04) isso foi atingido via cópia do diretório “ proc” relacionado aos processos sob o “cgroup” associado ao container e salvo em disco. Para relacionar o instantâneo a sua origem, usamos como nome do arquivo contendo o instantâneo da memória a combinação do hash da imagem e o hash do container como mostrado na Figura 2.

As técnicas forenses praticadas hoje estão voltadas para a obtenção da informação em sua totalidade, seja via cópia bit a

Fig. 2. Evidência salva - hash do container e imagem

bit, seja por remoção do hardware [15] [16]. Tais práticas tem levado ao crescente volume de dados que os investigadores tem que analisar. Há uma vertente na comunidade chamada "sniper forensics" onde se coleta e armazena o suficiente para a investigação. A solução proposta por este trabalho acompanha esta tendência, a questão foi definir a quantidade de dados "suficiente" para uma investigação. Decidimos que "suficiente" seria a quantidade necessária para descrever o sistema antes e depois do ataque. A idéia é implementar um log rotativo de instantâneos de memória cobrindo uma quantidade de tempo configurável, integrar a solução com algum sistema de detecção de ameaça de modo que, ao detectar um ataque, o log passa de rotativo a completo assim permitindo que se conheça o sistema antes e depois do ataque como mostrado na Figura 3.

Fig. 3. Janela deslizante de coleta de evidência



De modo a não violar a jurisdição de outros países ou a privacidade de outros usuários por causa do caráter multi-inquilino e multi-jurisdição das arquiteturas em núvem pública, a solução proposta por este trabalho foi o de armazenar a evidência em um local físico fora da nuvem utilizando como transporte conexão segura. Outro ponto importante é garantir a cadeia de custódia da evidência ou seja, garantir que a evidência não foi destruída, alterada ou acessada por qualquer pessoa. Assim a solução proposta por este trabalho usará de armazenamento físico fora da nuvem, o transporte será feito por TLS e o acesso a evidência será controlado.

Tendo a implementação sido bem sucedida conseguiremos analisar e identificar as formas de ataque enumeradas nos objetivos.

C. Implementação

Usando qual máquina, qual versão do virtual box, qual versão do docker, implementando qual container, como usei para identificar o id do processo ao container e assim, como associei o conteúdo da memória ao container e a imagem. Como implementei a janela deslizante.

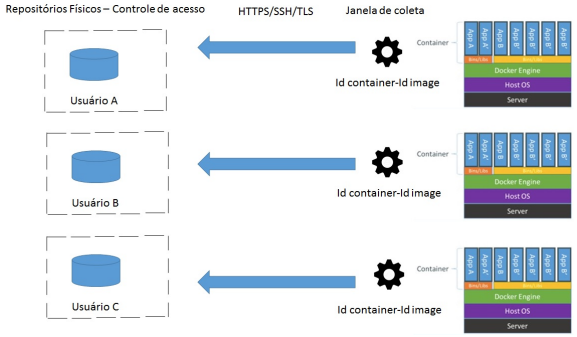
D. Limitações

Ameaças das quais estamos focando neste trabalho usam técnicas que permitem passar desapercebidas pelo processo de detecção de ameaças. Algumas delas são, adulteração da lista de processos ativos em uma máquina, se fazer passar por um processo válido ou se fazer passar por uma biblioteca válida [17]. Por isso, mesmo que haja uma integração com alguma forma de detecção de ameaça para a mudança do armazenamento de janela para o armazenamento total, acreditamos que ainda é necessária a capacidade de acionamento manual.

A solução esta focada em coletar informações de memória do espaço de memória do usuário assim, mesmo que ela ajude na investigação de ameaças que realizem manipulação direta dos objetos do Kernel (*D.K.O.M. - Direct Kernel Object Manipulation*) Kernel space no host não se beneficia da associação com o container.

A solução completa com todos os elementos descritos anteriormente pode ser visto na figura 4

Fig. 4. Solução completa



VI. Conclusion

Dado que o identificador de uma imagem e de um container são únicos, é possível associar o conteúdo da memória de um container a seu identificador de container e imagem, assim mesmo que as máquinas sejan instanciadas e deletadas a imagem que gerou aquela print de memória existe e pode ser reproduzido. Por causa da organização da memória, as posições absolutas não podem ser mantidas mas o conteúdo sim.

VII. Trabalhos futuros

Apesar de conseguirmos relacionar a evidência a sua fonte e reproduzir o processo, ainda precisamos verificar se coletar apenas a memória do container é o suficiente realizar uma análise e/ou encontrar uma forma segura de associar a memória do container com a do kernel da máquina hospedeira. Acreditamos estarmos no caminho certo pois, para a detecção das ameaças declaradas no inicio do documento, uma das premissas é a existência de uma cópia saudável da memória de um processo.

Acknowledgment

Beijo para a minha mae, meu pai e para a xuxa.

References

[1] AMAZON. *Amazon Media Room Press Release*. [S.l.], 2016. 2 p.

[2] KEYUN, R. et al. *Advances in Digital Forensics IV*. 7. ed. Orlando: [s.n.], 2011. 35–46 p. ISSN 1098-6596. ISBN 9788578110796.

[3] DYKSTRA, J.; SHERMAN, A. T. Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques. *Digital Investigation*, Elsevier Ltd, v. 9, n. SUPPL., p. S90–S98, 2012. ISSN 17422876.

[4] GEORGE, S.; VENTER, H.; THOMAS, F. Digital Forensic Framework for a Cloud Environment. In: CUNNINGHAM, P.; CUNNINGHAM, M. (Ed.). *IST Africa 2012*. Tanzania: Internation Information Management Corporation, 2012. p. 1–8. ISBN 9781905824342.

[5] 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*, v. 4, n. 1, p. 135–152, 2013. ISSN 20935374 (ISSN).

[6] DYKSTRA, J.; SHERMAN, A. T. Design and implementation of FROST: Digital forensic tools for the OpenStack cloud computing platform. *Digital Investigation*, Elsevier Ltd, v. 10, n. SUPPL., p. S87–S95, 2013. ISSN 17422876.

[7] REICHERT, Z.; RICHARDS, K.; YOSHIGOE, K. Automated forensic data acquisition in the cloud. *Proceedings - 11th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, MASS 2014*, p. 725–730, 2015.

[8] SANG, T. A log-based approach to make digital forensics easier on cloud computing. *Proceedings of the 2013 3rd International Conference on Intelligent System Design and Engineering Applications, ISDEA 2013*, p. 91–94, 2013.

[9] DEZFOULI, F. N. et al. Volatile memory acquisition using backup for forensic investigation. *Proceedings 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic, CyberSec 2012*, p. 186–189, 2012.

[10] DOLAN-GAVITT, B. et al. Virtuoso: Narrowing the semantic gap in virtual machine introspection. *Proceedings - IEEE Symposium on Security and Privacy*, p. 297–312, 2011. ISSN 10816011.

[11] 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, v. 11, p. S54–S62, 2014. ISSN 17422876.

[12] ZHANG, L.; ZHANG, D.; WANG, L. Live Digital Forensics in a Virtual Machine. In: *2010 Internation Conference on Computer Application and System Modelling (ICCASM 2010)*. [S.l.: s.n.], 2010. v. 6, p. 328–332.

[13] ALJAEDI, A. et al. Comparative Analysis of Volatile Memory Forensics. *IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT) and IEEE International Conference on Social Computing (SocialCom)*, p. 1253–1258, 2011.

[14] BARBARA, D. *Desafios da perícia forense em um ambiente de computação nas nuvens*. [S.l.], 2014.

[15] SIMOU, S. et al. Cloud forensics: Identifying the major issues and challenges. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 8484 LNCS, p. 271–284, 2014. ISSN 16113349.

[16] BEM, D. et al. Computer Forensics - Past , Present and Future. *Journal of Information Science and Technology*, v. 5, n. 3, p. 43–59, 2008.

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

[18] SOUSA, F. R. C.; MOREIRA, L. O.; MACHADO, J. C. Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios. *II Escola Regional de Computação, Ceara, Maranhão, Piauí (ERCEMAPI)*, v. 1, n. EDUFPI, p. 150–175, 2009.

[19] CHARTERS, I.; SMITH, M.; MCKEE, G. The Evolution of Digital Forensics. In: *Techno Forensics 2008 Conference*. [S.l.: s.n.], 2008. p. 1–39.

[20] GRISPOS, G.; STORER, T.; GLISSON, W. Calm before the storm: the challenges of cloud computing in digital forensics. *International Journal of Digital Crime and Forensics*, v. 4, n. 2, p. 28–48, 2012. ISSN 1466640073.

[21] SHARMA, H.; SABHARWAL, N. Investigating the Implications of Virtual Forensics. *Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on*, p. 617–620, 2012.

[22] RAFIQUE, M.; KHAN, M. N. A. Exploring Static and Live Digital Forensics: Methods, Practices and Tools. *International Journal of Scientific & Engineering Research*, v. 4, n. 10, p. 1048–1056, 2013.

[23] QUICK, D.; CHOO, K. K. R. Impacts of increasing volume of digital forensic data: A survey and future research challenges. *Digital Investigation*, Elsevier Ltd, v. 11, n. 4, p. 273–294, 2014. ISSN 17422876.