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

A literatura voltada a análise forense na nuvem foi analisada a luz dos seguintes conceitos pertinentes a este trabalho.

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

Com a revisão constatou-se que referente a coleta de informações, a maioria dos trabalhos mais importantes na área como [7], [5], [6], [4] e [8] focam em coleta reativa pois ela acontece apenas após a ameaça ser detectada. Os processos de coleta descritos no trabalho são iniciados manual ou integrada a um mecanismo de detecção de intrusão. Em casos de memória volátil, tal forma de coleta de informação não consegue descrever como era a memória antes do ataque pois o processo só é acionado depois do ataque. A capacidade de saber como era a memória antes do ataque é descrita por autores como [17] como necessária para viabilizar a abordagem de coletar o mínimo necessário para realizar a investigação. A única proposta encontrada que leva tal necessidade em consideração é [9] mas propõe que o dado seja armazenado no próprio dispositivoo que pode levar a perda de informações importantes caso a máquina virtual seja despejada do pool e seus recursos liberados.

Ainda na coleta de informações, os autores [7], [4] sugerem a abordagem de forense ao vivo onde os dados são constantemente coletados. Os autores [5], [6], [8] adotam a estratégia de isolar e parar a máquina virtual para em seguida realizar o processo de coleta. Nos casos citados anteriomente, nenhum dos autores cobre o cenário onde é necessário evidências de uma máquina virtual que já foi despejada do pool e os recursos liberados. Atender este cenário é importante pois com as soluções em nuvem que escalam automaticamente, as evidências de uma máquina vítima de um ataque que foi despejada de um pool com a diminuição da demanda serão para sempre perdidas. Analisando a proposta de [5], parece ser possível cobrir o cenário mencionado anteriormente mas ele não dá detalhes da implementação suficientes para termos certeza.

B. Capacidade de reproduzir o processo e obter os mesmos resultados.

Referente a capacidade dos trabalhos de reproduzir o processo de coleta, nenhuma das propostas consegue reproduzir os mesmos resultados ao repetir o processo no cenário em que uma máquina virtual é despejada da nuvem e seus recursos liberados pois todas elas dependem da existência da máquina virtual para a realização da coleta. Analisando cuidadosamente a proposta de [4] parece que é possível mas o autor não dá detalhes de implementação suficientes para termos certeza.

C. Não violar privacidade ou jurisdição das partes não envolvidas na investigação.

No armazenamento das informações coletadas, [7], [4], [5] e [6] usam estratégia de armazenamento fora da nuvem para evitar o problema de jurisdição e violação de privacidade de outros usuários no processo da coleta. No trabalho de [8] e um caso específico de [4] há uma dependência da cooperação do provedor de serviços de nuvem para conseguir as informações necessárias à investigação. Depender do provedor de serviços de nuvem é considerada uma estratégia fraca pela comunidade forense pois o foco do provedor de núvem é garantir a continuidade do serviço não a coleta de evidências.

D. Garantir a cadeia de custódia da evidência.

Na garantia da cadeia de custódia apenas [8] tenta resolver a questão mas toma cuidados apenas para garantir que a evidência não foi destruída ou alterada. Ele faz isso através de cálculo de hashing mas não explica como impede o acesso não autorizado à evidência. As propostas dos outros autores estão focadas apenas no aspecto técnico da coleta, nenhum deles menciona garantia de custódia apenas que as evidências são coletadas de forma ”forensicamente aceitável”.

Tabela comparativa das soluções

TABLE I
COMPARATIVO DE SOLUÇÕES

	Coleta é contínua?	Reproduz o processo sem a VM?	Garante cadeia de custódia?	Preserva jurisdição e privacidade?
[4]	✘	✘	✘	✓
[5]	✘	✘	✘	✓
[6]	✘	✘	✘	✓
[14]	✘	✘	✘	✓
[7]	✘	✘	✓	✓
[8]	✓	✓	✘	✓
[10]	✘	✘	✘	✓
[13]	✘	✘	✘	✓
[9]	✓	✘	✘	✓
[11]	✓	✘	✓	✓

V. SOLUÇÃO PROPOSTA

A. Objetivos

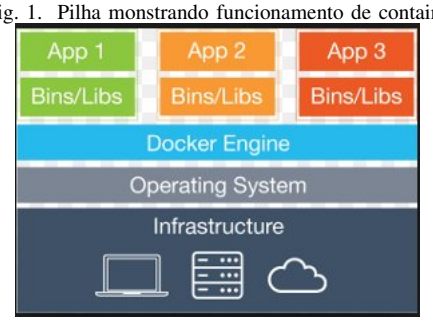
O presente proposta tem os seguintes objetivos:

- 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



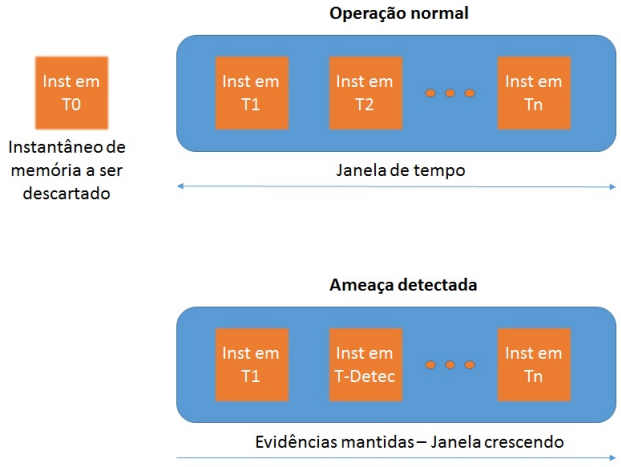
A solução proposta por este trabalho, para resolver o problema de associação da evidência a sua origem de modo que o processo seja reprodutível, pausa a execução do container e coleta um instantâneo da memória dos processos sob sua execução. Este processo é executado em intervalos de tempo conhecidos 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.

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

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

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 ar-

Fig. 3. Janela deslizante de coleta de evidência



mazenar 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

A implementação da solução foi realizada em um notebook intel I5 de 2.30Mhz e 4Gb de RAM com sistema operacional de 64 bits. Nele, usando Oracle Virtual Box 5.0 criamos uma máquina virtual com 2 Gb de memória RAM emulando apenas 1 processador.

Na máquina virtual instalamos a versão 1.9.1 do Docker engine e 1.21 da API, criamos 3 containers, cada um rodando um nginx 1.0 em diferentes portas. Foi escrita uma aplicação em JAVA que descobre qual o PID associado a cada container e salva o **/proc/pid/numa_maps** em um arquivo cujo nome é **container_id-imagem_id-hora-minuto.mem**

A cópia e gravação do arquivo acontece da seguinte forma, a cada minuto a aplicação pausa o container em questão, tira uma cópia do numa_maps e a salva em um arquivo .mem. Em seguida verifica qual o arquivo .mem mais antigo em disco, se for mais velho que o tempo 't', o arquivo é descartado.

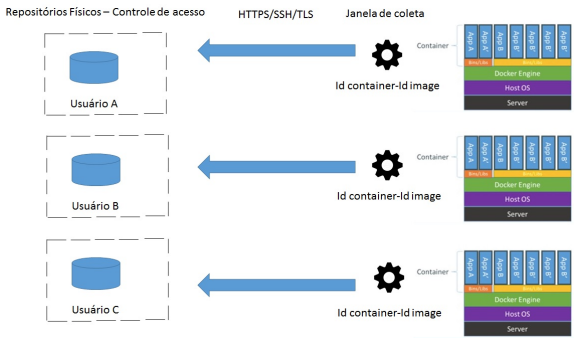
D. Limitações

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.