Hamilton Fonte II

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

São Paulo, Brasil

2016, v-0.1

Hamilton Fonte II

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

Projeto de pesquisa para a disciplina Metodolodia de Pesquisa Científica em Engenharia de Computação.

Universidade de São Paulo – USP

Escola Politécnica - Engenharia de Computação

Programa de Pós Graduação em Engenharia Elétrica - Mestrado

Orientador: Marcos Antonio Simplício Jr

São Paulo, Brasil

2016, v-0.1

Sumário

1 Introdução

MARCOS: Apresente aqui o contexto em que se insere o seu trabalho, sem se preocupar muito com o problema exato a ser resolvido ou como você irá fazê-lo.

1.1 Motivação

MARCOS: Apresente aqui as motivações do seu trabalho, deixando claro qual o problema que aparece no contexto em questão e o porquê dele ser importante, sem ainda se preocupar em como você pretende resolvê-lo.

Aumento do uso de soluções de virtualização e a implementação de arquiteturas em nuvem que escalam automáticamente (AMAZON, 2016) 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 de 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.

Para o escopo deste trabalho estamos considerando 4 tipos de ataques realizados diretamente na memória e que não deixam rastros no disco da máquina, todos baseados em injeção de código (CASE et al., 2014).

- **Injeção remota de bibliotecas** - Um processo malicioso força o processo alvo a carregar uma biblioteca em seu espaço de memória através de comandos do sistema operacional. A biblioteca existe fisicamente em alguma localização remota.
- **Injeção remota de código** - Um processo malicioso escreve código como uma sequência de bytes diretamente no espaço de memória de um processo alvo e força este último a executa-lo. O código pode por exemplo ser um script de shell
- **Injeção reflexiva de biblioteca** - Um processo malicioso escreve diretamente na memória de um processo alvo, como uma sequência de bytes, o código de uma biblioteca e força o processo alvo a executa-la. Nesta forma de ataque a biblioteca não existe fisicamente.
- **Injeção de processo vazio** - Um processo malicioso dispara uma instância de um processo legítimo no estado suspenso, a área do executável é liberada e realocada com código malicioso.

Do ponto de vista forense, praticantes e pesquisadores concordam aspectos de multi-inquilino e multi-jurisdição próprios soluções em nuvem figuram entre as principais dificuldades para coleta de evidência (KEYUN et al., 2011). 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 a obtenção da mesma (DYKSTRA; SHERMAN, 2012).

O crescente volume de dados das aplicações atuais deram aos investigadores em média de 6 a 12 meses de backlog para investigar (QUICK; CHOO, 2014), download de terabytes de dados leva horas para se realizar e requer a colaboração do provedor de nuvem. Mudar o paradigma de coleta tem sido proposto por pesquisadores e praticantes já há alguns anos (BIRK; WEGENER, 2011)(SANG, 2013) mas estas práticas precisam garantir a cadeia de custódia para que as evidências produzidas por ela sejam aceita em um processo legal.

1.2 Objetivos

MARCOS: Apresente aqui quais são os seus objetivos, ou seja, como você irá resolver o problema em questão.

- 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.

1.3 Justificativa

MARCOS: Apresente aqui o porquê da forma como você vai resolver o problema ser interessante, bem como possíveis alternativas e o porquê das alternativas não serem tão interessantes quanto a sua abordagem.

falar da tendência para o sniper forensics, falar não só de coleta mas de toda a cadeia de coleta e preservação da evidência

1.4 Organização do documento

O presente documento está organizado em capítulos, da seguinte forma. O capítulo 3 apresenta ...

2 Fundamentação Teórica

MARCOS: Discuta aqui os conceitos básicos necessários para entender sua solução e os trabalhos relacionados

A prática forense esta relacionado fundamentalmente ao controle da evidência

Conceito de isolamento da área e coleta da evidência.

Conceito de reprodutibilidade da coleta

Concento da cadeia de custódia e a garantia da idoneidade da prova https://qperito.com/2013/11/o-sr-perito-explicar-o-que-e-cadeia-de-custodia/

3 Análise forense na nuvem: requisitos e trabalhos relacionados

MARCOS: Precisa de um texto introdutório aqui. Estou deixando uma sugestão inicial

Neste capítulo são discutidos os principais requisitos para soluções de análise forense na nuvem. Com base nesses requisitos, são então apresentadas algumas das principais soluções relacionadas à presente proposta, bem como suas limitações.

3.1 Requisitos

MARCOS: Discuta aqui as métricas que você vai utilizar para a comparação e o porquê delas serem relevantes. Isso define os requisitos para soluções relacionadas, e permite você analisá-las e justificar a sua solução. Até o momento não encontrei principais requisitos, há requisitos da forense mas eles conflitam com as pectos da núvem.

Coleta contínua, presevação da evidência volátil, independência de VM, conhece o contexto, cadeia de custódia

3.2 Revisão da literatura

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

MARCOS: Maus hábitos detectados na sua forma de escrita: excesso de sujeito oculto (veja os “quem?” e “o que” abaixo para perceber que há o uso de sujeito oculto reduz a clareza nesses pontos); uso incorreto de “esse vs este” (sempre que você falar “este”, é o *seu* trabalho; se quiser se referir ao trabalho que acaba de ser mencionado, diga *esse*)

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

Referente a coleta de informações, os autores (REICHERT; RICHARDS; YOSHIGOE, 2015), (POISEL; MALZER; TJOA, 2013), (DYKSTRA; SHERMAN, 2013), (GEORGE; VENTER; THOMAS, 2012) e (SANG, 2013) focam em coleta "após o fato"pois ela acontece apenas após a intrusão ser detectada. Os processos de coleta descritos nos trabalhos são 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. A capacidade de saber como era a memória antes do fato é descrita por (CASE et al., 2014) como necessária para viabilizar a abordagem de coletar o suficiente para realizar a investigação pois permite comparar dois instantâneos de memória e minimizar o volume coletado antes do fato. A única proposta encontrada que leva tal necessidade em consideração é (DEZFOULI et al., 2012) mas propõe que o dado seja armazenado no próprio dispositivo porém essa abordagem não é aplicavel ao cenário em nuvem pois leva a perda de informações importantes caso a máquina virtual seja despejada e seus recursos liberados.

Ainda na coleta de informações, os autores (REICHERT; RICHARDS; YOSHIGOE, 2015), (GEORGE; VENTER; THOMAS, 2012) sugerem a abordagem de forense ao vivo onde os dados são constantemente coletados sem distinção do antes ou depois do fato e os autores (POISEL; MALZER; TJOA, 2013), (DYKSTRA; SHERMAN, 2013), (SANG, 2013) adotam a estratégia de isolar e parar a máquina virtual para em seguida realizar o processo de coleta. Nas duas estratégias citadas anteriormente, o problema do grande volume de informações coletadas não é abordado pelo autores nem o cenário onde é necessário coletar evidências de uma máquina virtual que já foi despejada do pool e os recursos liberados. Atender este último 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 (POISEL; MALZER; TJOA, 2013), parece ser possível cobrir o cenário mencionado mas ele não dá detalhes da implementação suficientes para termos certeza.

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

A reprodutibilidade do processo de coleta é uma dos requisitos para garantir a cadeia de custódia da evidência. Cadeia de custódia esta relacionado a credibilidade e ter dois analistas reproduzindo o processo de coleta de memória chegando ao mesmo conjunto de evidências tem um peso muito forte em termos de credibilidade. Neste tópico, 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 a proposta de (GEORGE; VENTER; THOMAS, 2012) parece que é possível mas o autor não dá detalhes de implementação suficientes para termos certeza.

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

No caso das soluções em nuvem, não é possível remover o hardware para análise pois ele contem informações de vários usuários, alguns dos quais não estão envovidos na investigação em curso, fazê-lo levaria a violações de privacidade o que diminui a credibilidade da evidência. A maioria dos autores resolve este problema adequadamente e podemos listar duas estratégias usadas. Os autores (REICHERT; RICHARDS; YOSHIGOE, 2015), (GEORGE; VENTER; THOMAS, 2012), (POISEL; MALZER; TJOA, 2013) e (DYKSTRA; SHERMAN, 2013) usam estratégia de coletar dados pertinentes a investigação e armazena-los fora da nuvem enquanto que (SANG, 2013) e um caso específico de (GEORGE; VENTER; THOMAS, 2012) dependem 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.

3.2.4 Garantir a cadeia de custódia da evidência.

Na garantia da cadeia de custódia apenas (SANG, 2013) aborda a questão mas toma cuidados somente para garantir que a evidência não foi destruída ou alterada através do cálculo de hashing da mesma mas não explica como impede o acesso não autorizado. 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

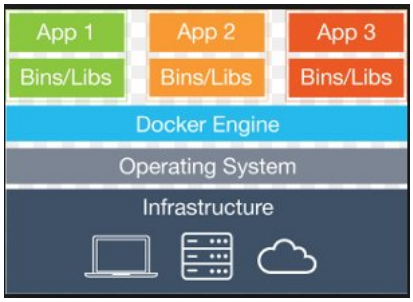
Tabela 1 – Comparativo de soluções

	Coleta é contínua?	**Reproduz o processo sem a VM?**	**Garante cadeia de custódia?**	**Preserva jurisdição e privacidade?**
(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)	✓	✘	✓	✓

4 Descrição da Solução

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.

Figura 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.

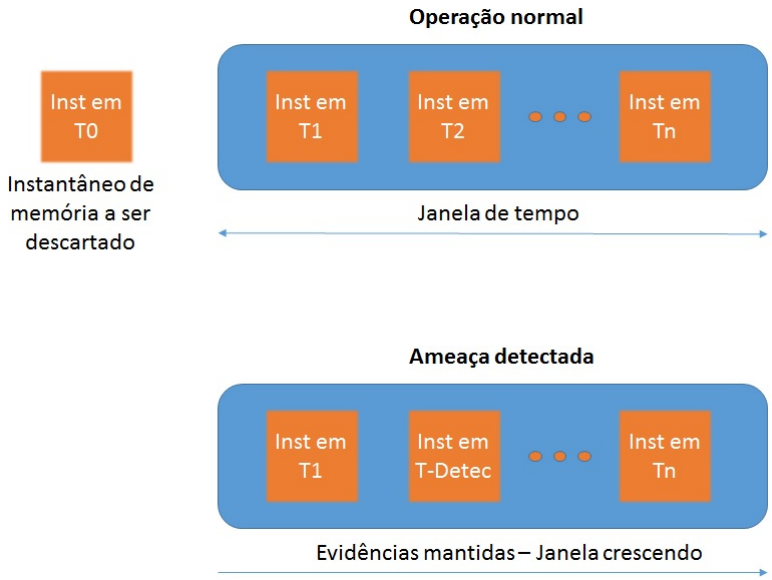
Figura 2 – Evidência salva - hash do container e imagem

```
-rw-r--r-- 1 root root 13376 Jul 18 22:40 4bd952884935d8042113340013029042977 8acc85df0ed7366e23a9d19425d1d-8fa80c6dba11002f45c835254343bce274fa
27e1136708a0e4cb13ecf57d6b53-3502-18_07_2016_10_40.mem
-rw-r--r-- 1 root root 12782 Jul 18 22:40 4bd952884935d8042113340013029042977 8acc85df0ed7366e23a9d19425d1d-8fa80c6dba11002f45c835254343bce274fa
27e1136708a0e4cb13ecf57d6b53-3520-18_07_2016_10_40.mem
-rw-r--r-- 1 root root 12782 Jul 18 22:40 4bd952884935d8042113340013029042977 8acc85df0ed7366e23a9d19425d1d-8fa80c6dba11002f45c835254343bce274fa
27e1136708a0e4cb13ecf57d6b53-3521-18_07_2016_10_40.mem
-rw-r--r-- 1 root root 12782 Jul 18 22:40 4bd952884935d8042113340013029042977 8acc85df0ed7366e23a9d19425d1d-8fa80c6dba11002f45c835254343bce274fa
27e1136708a0e4cb13ecf57d6b53-3522-18_07_2016_10_40.mem
-rw-r--r-- 1 root root 12782 Jul 18 22:40 4bd952884935d8042113340013029042977 8acc85df0ed7366e23a9d19425d1d-8fa80c6dba11002f45c835254343bce274fa
27e1136708a0e4cb13ecf57d6b53-3523-18_07_2016_10_40.mem
-rw-r--r-- 1 root root 13354 Jul 18 22:40 6f7e69cc438812334817a9211236b36c3b71b0e8dd606046631ee1c8625e142d-8fa80c6dba11002f45c835254343bce274fa
27e1136708a0e4cb13ecf57d6b53-3426-18_07_2016_10_40.mem
```

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 (SIMOU et al.,

2014) (BEM et al., 2008). 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.

Figura 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.

Limitações da solução

Ameaças das quais estamos focando neste trabalho usam técnicas que permitem

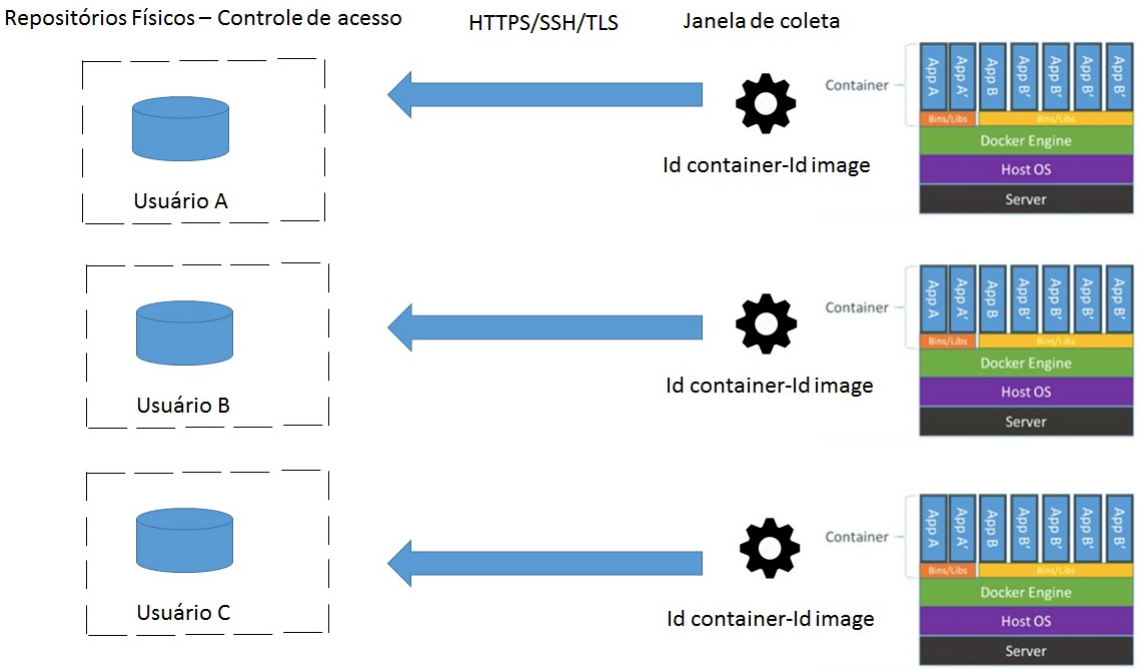
passar desapercebidas pelo processo de detecção de ameaças. Algumas delas são, adulteração da lista de processos ativos em uma máquina, se fazer passar por um processo válido ou se fazer passar por uma biblioteca válida (CASE et al., 2014). Por isso, mesmo que haja uma integração com alguma forma de 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.

Esquema da solução

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

Figura 4 – Solução completa



5 Análise dos Resultados

Aqui a análise dos resultados (QUICK; CHOO, 2014)

Referências

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. Citado na página 10.

AMAZON. *Amazon Media Room Press Release*. [S.l.], 2016. 2 p. Citado na página 3.

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. Disponível em: <http://dx.doi.org/10.1016/j.diin.2014.03.007>. Citado na página 10.

BARBARA, D. *Desafios da perícia forense em um ambiente de computação nas nuvens*. [S.l.], 2014. Citado na página 10.

BEM, D. et al. Computer Forensics - Past , Present and Future. *Journal of Information Science and Technology*, v. 5, n. 3, p. 43–59, 2008. Citado na página 12.

BIRK, D.; WEGENER, C. Technical Issues of Forensic Investigations in Cloud Computing Environments. *2011 Sixth IEEE International Workshop on Systematic Approaches to Digital Forensic Engineering*, p. 1–10, 2011. Disponível em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6159124>. Citado na página 4.

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. Citado 3 vezes nas páginas 3, 8 e 13.

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. Citado 2 vezes nas páginas 8 e 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. Citado na página 10.

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. Disponível em: <http://dx.doi.org/10.1016/j.diin.2012.05.001>. Citado na página 4.

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. Disponível em: <http://dx.doi.org/10.1016/j.diin.2013.06.010>. Citado 4 vezes nas páginas 7, 8, 9 e 10.

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. Citado 4 vezes nas páginas 7, 8, 9 e 10.

KEYUN, R. et al. *Advances in Digital Forensics IV*. 7. ed. Orlando: [s.n.], 2011. 35–46 p. ISSN 1098-6596. ISBN 9788578110796. Citado na página 4.

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). Disponível em: <http://www.scopus.com/inward/record.url?eid=2-s2.0-84885399460{&}partnerID=40{&}md5=0e332690d4cb1f01934b540b53>. Citado 4 vezes nas páginas 7, 8, 9 e 10.

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. Disponível em: <http://dx.doi.org/10.1016/j.diin.2014.09.002>. Citado 2 vezes nas páginas 4 e 14.

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. Citado 4 vezes nas páginas 7, 8, 9 e 10.

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. Citado 5 vezes nas páginas 4, 7, 8, 9 e 10.

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. Citado na página 12.