# Uma Avaliação Comparativa dos Mecanismos de Segurança nas Ferramentas OpenStack, OpenNebula e CloudStack

**Demétrius Roveda**[1], **Adriano Vogel**[1], **Samuel Souza**[2], **Dalvan Griebler**[1,3]

[1] Laboratório de Pesquisas Avançadas para Computação em Nuvem (LARCC), Faculdade Três de Maio (SETREM) – Três de Maio – RS – Brasil

[2]Computer Science Department, Binghamton University – Binghamton, NY, EUA

[3]Programa de Pós-Graduação em Ciência da Computação, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS) – Porto Alegre – RS – Brasil

{*roveda.demetrius,adrianovogel03*}*@gmail.com, samuel@samuelsouza.com*
*dalvan.griebler@acad.pucrs.br*

***Abstract.*** *The IaaS service model is gaining attention due its importance to the cloud computing environment, it is responsible for simplifying the access and the management of high-end processing and storage systems, besides being the base that allows the outsourcing of upper layers, PaaS and Saas. The IaaS cloud tools are responsible for controlling the virtual infrastructure as well the environment security, which is an important characteristic for cloud applications, once the system can be integrated with public clouds through the Internet. In this paper, the goals are evaluate and compare the security layer, from the administrator point of view, of three open source IaaS tools: OpenStack, OpenNebula and Cloudstack. Considering the security layer from Dukaric taxonomy, the results shown that all the tools have a equivalent security level, however, there are evidence that not all the security features found in the tools fits in the taxonomy description.*

**Keywords**: Cloud Security; Cloud Computing; Infrastructure as a Service;

***Resumo.*** *O modelo de serviço IaaS vem tendo bastante atenção devido a sua importância para o ambiente de computação em nuvem, pois simplifica o acesso e o gerenciamento de ambientes com grande capacidade de processamento e armazenamento, além de ser a base que permite a terceirização das camadas superiores, PaaS e SaaS. As Ferramentas de gerenciamento de IaaS são responsáveis por controlar a infraestrutura virtual bem como a segurança do ambiente, que é extremamente importante para a nuvem, uma vez que um sistema pode se integrar com a nuvem publica através da Internet. Nesse artigo, o objetivo é avaliar a camada de segurança do ponto de vista do administrador das ferramentas de IaaS: OpenStack, OpenNebula e CloudStack. Considerando a camada de segurança da taxonomia proposta na literatura, os resultados demostram que todas as ferramentas possuem um nível de segurança similar, porém ficou evidenciado também que nem todas as soluções de segurança encontradas nas ferramentas se encaixam nas descrições da taxonomia.*

**Palavras Chaves**: Segurança em Nuvem; Computação em Nuvem; Infraestrutura como um Serviço;

## 1. Introdução

A computação em nuvem surgiu oferecendo recursos computacionais na forma de serviços através da internet e se tornou uma alternativa competitiva, trazendo diversas vantagens (eficiência energética, flexibilidade, elasticidade, baixo investimento inicial, entre outros) [Buyya et al. 2013]. Basicamente, ela é composta por três modelos de serviços, infraestrutura como serviço (IaaS), que provê recursos de *hardware* (processamento, memória, armazenamento, rede) para as camadas superiores, plataforma como serviço (PaaS) e *software* como serviço (SaaS) [ISO JTC1/SC38 Technical Report 2014].

As ferramentas de gerenciamento de IaaS abstraem a complexidade computacional a nível de *hardware* através da virtualização e oferecem uma interface para gerenciamento dos recursos computacionais, deixando transparente as atividades de *hardware* para as camadas superiores, facilitando a interação com os usuários [Zhou et al. 2010]. O desafio deste trabalho é fazer uma análise da segurança de três ferramentas de código aberto, do ponto de vista do administrador do ambiente (administrador/sistema), discutindo desafios e soluções existentes neste contexto. Por outro lado, as outras visões disponíveis que poderão ser exploradas no futuro, são elas: usuário/sistema, que é a forma com que o usuário interage com o sistema; e sistema/sistema, que é a forma como os módulos do sistema interagem com o sistema ou entre outros módulos.

A necessidade deste estudo foi percebida em pesquisas anteriores que analisaram exclusivamente a camada de gerenciamento em ferramentas para implantação de ambientes de nuvens privadas [Vogel et al. 2016, Roveda et al. 2015a, Roveda et al. 2015b, Thome et al. 2013]. Tais pesquisas foram relacionadas com [Dukaric and Juric 2013], onde foi proposta uma taxonomia conceitual para ferramentas de gerenciamento de IaaS, classificando em 7 camadas (segurança, abstração de recursos, serviços de valor agregado, controle, serviços de núcleo, suporte e gerenciamento), que são componentes relevantes para um funcionamento adequado. No entanto, a camada de segurança é vertical e afeta a nuvem como um todo.

Devido a importância que a segurança tem em ambientes de nuvem, onde clientes utilizam o mesmo ambiente, a taxonomia proposta não oferece uma análise tão abrangente, que leve em conta todos os itens propostos na teoria da segurança da informação, estes, fatores de grande impacto na segurança do ambiente. Outro fator que impacta diretamente na segurança e merece atenção, é a integração da nuvem privada com a nuvem pública, pois nela os usuários compartilham a mesma infraestrutura, e pode ser um risco em potencial, comprometendo o ambiente. As ferramentas analisadas possuem recursos que fazem essa integração, geralmente são através de APIs, estas devem possuir um alto nível de segurança, para que a troca de informações entre as nuvens ocorra de forma segura.

Ter uma percepção do nível de segurança das ferramentas é uma tarefa complexa, devido às abstrações de software que existem no modelo IaaS (virtualização, sistema operacional e sistema de gerenciamento de IaaS) e as características físicas do *data center* (controle de acesso físico, recuperação de desastres, políticas de segurança). Todos esses fatores contribuem para que o ambiente de nuvem possua algum nível de segurança. As ferramentas de IaaS que possuem um maior suporte a tecnologias, de certa forma são mais flexíveis e integram mais sistemas heterogêneos, do ponto de vista da segurança, essa interação é uma potencial vulnerabilidade, uma vez que quando há interações entre

sistemas, essas interfaces de comunicação ficam maiores, e mais difícil de gerenciá-las, sendo assim, andando em uma direção oposta à segurança do ambiente.

Em ambientes de nuvem, a utilização de boas práticas em TI (Itil v3 ou COBIT) agregam valor ao negócio, através da padronização de processos e tarefas, gerenciamento de riscos entre outros benefícios. Na computação em nuvem, o gerenciamento de riscos é um fator de grande importância, pois é possível mensurar o impacto no ambiente caso alguma vulnerabilidade do sistema seja explorada por uma ameaça. Um exemplo disso é: uma ameaça (pessoa mal intencionada) é considerado qualquer coisa que possa explorar uma vulnerabilidade (exemplo: porta aberta em *firewall*), e através da probabilidade dessa pessoa mal intencionada explorar essa vulnerabilidade no *firewall*, é possível mensurar o tamanho risco. Este risco está diretamente ligado ao valor do ativo (informação da organização), e através disso pode-se calcular o impacto dele para o ambiente.

Visto que grande parte das aplicações que rodam na nuvem são críticas, uma definição da estrutura dos componentes de segurança facilitaria na identificação de vulnerabilidades e prevenção contra ataques. Através desses fatores, este artigo tem como objetivo analisar as ferramentas de IaaS OpenStack, OpenNebula e CloudStack iniciando pela taxonomia de [Dukaric and Juric 2013] e, posteriormente, através de uma análise, avaliar e discutir características de segurança que deveriam estar presentes nas ferramentas para o administrador de IaaS. Contudo, as contribuições deste estudo são:

- Uma extensão dos estudos da taxonomia na camada de segurança proposta por [Dukaric and Juric 2013], elencando aspectos importantes que não foram considerados.
- Inclusão da ferramenta CloudStack nos estudos da taxonomia e uma análise das características de segurança (não foi considerada por [Dukaric and Juric 2013]).
- Avaliação e medição dos mecanismos de segurança em três ferramentas de nuvem privada (OpenStack, OpenNebula e CloudStack).

O artigo está organizado em 7 seções. Na subseção 2.1 é visto os atributos da segurança da informação. Na subseção 2.2 são apresentadas as 3 ferramentas de gerenciamento de IaaS (OpenStack, OpenNebula e CloudStack). Já na subseção 2.3, é evidenciada a taxonomia utilizada na comparação seguida dos itens da camada de segurança. Na seção 3 é relacionado o presente estudo com demais trabalhos encontrados na literatura. A seção 4 apresenta a comparação das ferramentas, seguida da seção 5, onde é discutido os resultados alcançados. Finalizando, na seção 6 são realizadas as conclusões do presente estudo e indicado trabalhos futuros.

## 2. Base Teórica

Essa seção contém uma base teórica, que aborda atributos da segurança da informação, ferramentas de gerenciamento de IaaS, bem como a taxonomia e itens da camada de segurança.

### 2.1. Atributos da Segurança da Informação

A segurança da informação é um conceito que se baseia em princípios, esses que visam conservar e proteger informações bem como sistemas de informações. Organizações tratam informações como ativos, que possuem um grande valor. Na computação em nuvem,

empresas prestadoras de serviços de nuvem são totalmente responsáveis pelos dados das organizações que utilizam esse tipo de serviço, entre esses dados podem estar informações críticas para as organizações, e um vazamento delas pode acarretar em um grande impacto para o negócio de ambas as partes. Casos desse tipo ainda envolvem questões legais e contratuais, podendo haver multa para a prestadora de serviço. Conforme [Bishop 2005], os princípios da segurança da informação estão organizados da seguinte forma:

- **Confidencialidade** é o atributo que refere-se em apenas pessoas autorizadas terem acesso ou saber da existência de determinados ativos. Uma das características mais importante da confidencialidade em ambientes de nuvem é a segregação de dados entre os usuários. Essa segregação também é aplicada na comunicação do ambiente de cada usuário (VLAN). Dentre várias formas de prover confidencialidade, os métodos mais utilizados são autenticação, controle de acesso de usuários e criptografia de informações.

- **Integridade** é o atributo que rege a exatidão de um ativo, que foi modificado de formas aceitáveis, por pessoas ou processos autorizados e as informações devem ser inteiramente consistentes. Na computação em nuvem, isso é visto como garantias de que os dados dos usuários são alterados apenas por pessoas autorizadas e em casos de desastres, se a informação não está de alguma forma inconsistente. Através do controle de acesso, algoritmos de coerência para base de dados replicadas e algoritmos de hash (que verificam se os dados foram alterados), os administradores de IaaS conseguem gerenciar esses fatores que na computação em nuvem é visto como um grande desafio.

- **Disponibilidade** é o atributo que define se o ativo está acessível pelas partes autorizadas em momentos apropriados. Na computação em nuvem, a disponibilidade é o princípio mais sensível pelo usuário de IaaS, e é um fator importante para a continuidade do negócio. Para administradores de IaaS, isso é tratado como questões de contingência do *data center*, como planos de contingência, redundância de links de acesso e de equipamentos.

### 2.2. Ferramentas de Gerenciamento de IaaS

Ferramentas de IaaS de código aberto são muito utilizadas na implantação de nuvens privadas. Essas são responsáveis pelo monitoramento e gerenciamento seguro dos recursos da nuvem [Vogel et al. 2016, Thome et al. 2013]. Em demandas específicas, o nível de controle bem como de recursos avançados (suporte para a venda de serviços ou pagamento por demanda) pode ser pobre ou inexistente. Dessa forma, é necessário o uso de ferramentas específicas para gerenciamento de nuvem [Shroff 2010]. Dentre diversas ferramentas de IaaS de código aberto, foram escolhidas OpenStack pela resiliência e aceitação corporativa, OpenNebula pela simplicidade de implantação e CloudStack pela eficiência no gerenciamento dos recursos.

- **OpenStack** é referência em ferramentas de IaaS de código aberto, devido a sua robustez e adaptividade, suportando diversas tecnologias [OpenStack 2015]. A OpenStack possui "Stacks" que são *StakeHolders* (investidores externos) nesse projeto, onde investem financeiramente no desenvolvimento e aprimoramento de recursos, que funcionam melhor com essa ferramenta. Devido a sua arquitetura,

ela é uma ferramenta modular, com mais de 40 componentes disponíveis para implementação. Cada um desses componentes é uma API e todas elas se comunicam através do mensageiro *RabbitMQ*.
Por outro lado, a sua implementação é mais complexa que a da ferramenta OpenNebula [Maron et al. 2014].

- **OpenNebula** é uma ferramenta de código aberto que surgiu em 2008 como uma solução para gerenciamento de IaaS de nuvem privada, pública e híbrida. A arquitetura da ferramenta é bem integrada, sem muitos componentes, o que reflete na fácil implantação do ambiente. O OpenNebula foi criado especialmente para gerenciar nuvens privadas, pois seus recursos são simplificados, comparando com as outras ferramentas de código aberto do mesmo segmento. Alguns recursos avançados de rede (criação de VLANs, *Firewall*, entre outros) só estão disponíveis com a utilização de recursos de terceiros (OpenVSwitch) [OpenNebula 2015]. Esses recursos, em nuvens públicas, são essenciais para o gerenciamento e segurança do ambiente.

- **CloudStack** também é de código aberto e pode gerenciar nuvens públicas, privadas e híbridas. A arquitetura da ferramenta também é bem integrada, e não possui muitas APIs (3). Ela é uma ferramenta flexível, pode-se facilmente implementar uma nuvem privada, mas também atende implementações robustas e escaláveis. A interface gráfica é intuitiva, o que facilita bastante o gerenciamento dos recursos virtuais. Através dela é possível executar comandos avançados (recuperação de máquinas virtuais (VMs)) sem precisar acessar alguma interface de linha de comando (CLI), que nativamente não é suportada, mas que está disponível através de recursos de terceiros (CLI *CloudMonkey*) [CloudStack 2015].

### 2.3. Taxonomia Conceitual de Segurança em IaaS

A Figura 1 ilustra a taxonomia proposta por [Dukaric and Juric 2013][1], sendo itens necessários para uma ferramenta de IaaS ser considerada "completa". Essas camadas estão divididas em sete blocos. Na base da taxonomia tem-se a camada de abstração de recursos que é a mais próxima da virtualização e oferece recursos computacionais (CPU, memória, volumes e redes) como serviços na nuvem. Na camada de núcleo de serviços é onde os serviços e clientes são controlados, identificados e ainda alguns serviços são implementados (repositório de imagens, medição e cobrança). Outro aspecto importante presente nessa camada é o escalonador de recursos.

A camada de gerenciamento é responsável por controlar as operações e usuários de uma nuvem, principalmente através de grupos e do monitoramento. Conta ainda com interfaces (gráficas, APIs ou CLIs) para que ocorra interações entre clientes e recursos. Outros aspectos do gerenciamento de nuvem que podem ser destacados são a elasticidade e orquestradores, para controle otimizado da infraestrutura. A camada de suporte busca oferecer serviços adicionais como banco de dados, transferências e mensagens entre os servidores e nuvens.

Na camada de controle é onde as políticas da nuvem são implementadas. Os acordos de nível de serviço garantem critérios a serem seguidos e a medição controla a

[1] A taxonomia é um *towards* (proposta em direção, não definitiva)

utilização dos recursos para o processo de pagamento. A camada de serviços agregados oferece funcionalidades adicionais como migração de VMs, tecnologias para alta disponibilidade e portabilidade.

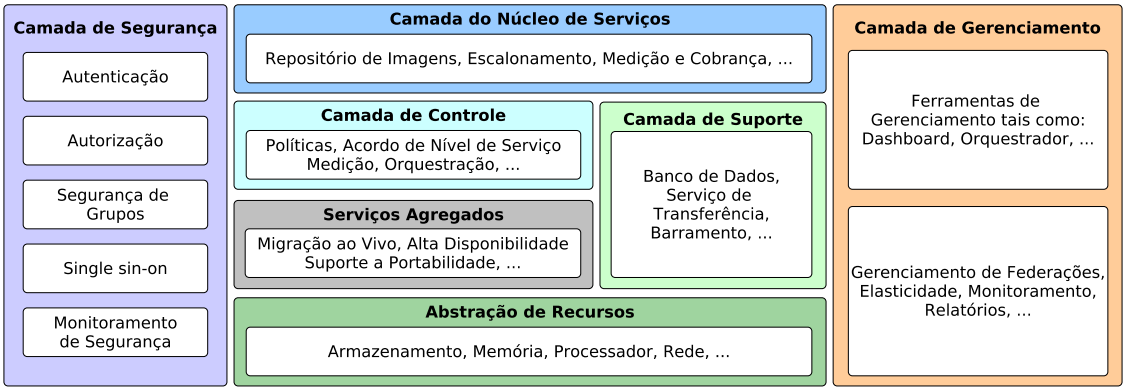


**Figura 1. Taxonomia conceitual de IaaS. Adaptado de [Dukaric and Juric 2013]**

Por último, a camada de segurança está sendo analisada neste trabalho, e possui os seguintes itens:

- **Autenticação** é a garantia que as partes de uma comunicação, são realmente quem afirmam ser. O mecanismo de autenticação mais comum é através de usuário/senha.
- **Autorização** é um processo que ocorre após a autenticação. Está relacionado aos privilégios de cada usuário em um determinado sistema. Os privilégios dos usuários podem ser herdados dos grupos no qual eles pertencem.
- **Grupos de Segurança (Isolamento de redes)** Em ambientes de nuvem IaaS, as redes virtuais dos usuários compartilham a mesma infraestrutura física. As VMs de um grupo ou usuário devem estar isoladas, de tal forma que nenhum usuário tenha algum tipo de acesso em uma VM pertencente a outra rede virtual.
- **Single sing-on** é um mecanismo que através de uma única autenticação (usuário/ senha, *token*), o usuário terá acesso a diversos sistemas que estejam integrados com esse recurso.
- **Monitoramento de Segurança** pode abranger inúmeros itens, mas no contexto de [Dukaric and Juric 2013], ele prevê o monitoramento das redes virtuais, logs de auditoria, o monitoramento de VMs, entre outros recursos que detectem eventos desconhecidos no ambiente de IaaS.

O gerenciamento da segurança de um ambiente de IaaS é complexo para o administrador. Em uma visão vertical, começa na segurança física do ambiente, onde apenas pessoas autorizadas devem ter acesso. Logo, é seguido do isolamento de recursos, pois os usuários compartilham o mesmo *hardware*. Nesse nível da segurança, o virtualizador é diretamente dependente e/ou responsável. Logo acima, vem a segurança do ambiente virtual, que refere-se ao suporte das ferramentas na utilização de tecnologias de integração com outras nuvens, gerenciado pela ferramenta de IaaS. Outro fator é o acesso do usuário ou administrador de IaaS até a ferramenta, que também pode ser feito através da internet, sendo efetivamente relevante no contexto da segurança. Ainda, a camada de segurança proposta possui itens sobrepostos (SSO e Autorização) e itens faltantes (Atributos da Segurança da Informação).

**Tabela 1. Trabalhos Relacionados**

| **Método** | ***Dukaric 2013*** | ***Dawoud 2013*** | ***Vaquero 2011*** | ***Albaroodi 2013*** | ***Este Trabalho*** |
|---|---|---|---|---|---|
| Quantitativo | | | | | X |
| Comparativo | X | | | | X |
| Qualitativo | X | X | X | X | X |
| **Desenvolvimento** | ***Dukaric 2013*** | ***Dawoud 2013*** | ***Vaquero 2011*** | ***Albaroodi 2013*** | ***Este Trabalho*** |
| Survey | X | | X | | |
| Levantamento | | X | X | X | X |
| Análise | | X | | X | X |
| **Resultados** | ***Dukaric 2013*** | ***Dawoud 2013*** | ***Vaquero 2011*** | ***Albaroodi 2013*** | ***Este Trabalho*** |
| Modelo de Seg. | | X | X | | |
| Taxonomia | X | | | | |
| Análise de Ferr. | | | | X | X |
| Classificação | X | | | | |

## 3. Trabalhos Relacionados

Devido a importância da segurança e privacidade em ambientes de nuvem, a literatura possui vários estudos relacionados com segurança da informação.

O desenvolvimento feito nos estudos relacionados também foi revisado. Nesse contexto, alguns trabalhos apresentam um *survey* [Dukaric and Juric 2013], que é uma pesquisa aprofundada de aspectos de segurança e/ou levantamento de partes relacionadas. Também são encontradas análises de segurança em ambientes ou ferramentas, como no estudo de [Albaroodi et al. 2014]. Os resultados dos trabalhos foram segmentados em tópicos relativos a cada estudo, segregando os trabalhos por: modelo de segurança (pode ser proposto como uma boa prática a ser seguida e que geralmente é idealizada a partir de um *survey*), análise de ferramentas (abordagem prática de aspectos), e por taxonomia (geralmente é uma proposta conceitual para se classificar ou comparar tecnologias e seus aspectos, sendo muito utilizada em pesquisas quantitativas).

No trabalho de [Dawoud et al. 2010] é proposto um modelo de segurança para IaaS, abrangendo a segurança de maneira mais completa e específica que a taxonomia do Dukaric. Os autores também identificam diferentes componentes de segurança dentro do modelo de computação em nuvem e apontam as principais ameaças encontradas bem como possíveis soluções. Por exemplo; *cloud Software* pode estar suscetível a ataques XML e uma possível solução para esse problema é a assinatura dos arquivos XML e/ou a sua criptografia. Uma diferença para o nosso trabalho é que em [Dawoud et al. 2010] os autores tratam a computação em nuvem de uma maneira genérica e não buscam avaliar ferramentas específicas. Em [Dawoud et al. 2010], os autores buscam identificar ameaças e trazer soluções e esta ainda é uma ação fora do escopo do nosso trabalho. Mas [Dawoud et al. 2010] serve pra demonstrar a complexidade de sistemas de *cloud*, os vários níveis envolvidos e a necessidade de expansão da taxonomia proposta por Dukaric a fim que sejam comportados todos os elementos de segurança que este tipo de arquitetura requer.

O estudo de [Vaquero et al. ] avalia os riscos de segurança em sistemas de *cloud computing* que compartilham máquinas de hospedagem, especificamente para serviços de IaaS. Realizam um survey sobre as principais ameaças, soluções e os riscos envolvidos nesse tipo de sistema. O estudo de [Vaquero et al. ] atua especificamente a nível de máquinas virtuais, trabalhando com o ciclo de vida das principais operações em ambientes virtualizados, bem como com as ameaças que estas ações introduzem. É um trabalho

relativamente abrangente, mas se limita a 7 ameaças, caracterizadas pelos autores como as principais. Assim como em [Dawoud et al. 2010], as ameaças são genéricas e não podem ser atribuídas a ferramentas específicas, o que pode causar modificações a cada diferente uso. Além do mais, nosso trabalho foca mais em soluções de um nível acima do gerenciamento de Máquinas virtuais (VMM). Os resultados de [Vaquero et al. ] são interessantes e auxiliam a mapear as diferentes vulnerabilidades em sistemas de *cloud* e facilita a proposição de um modelo de adversário para testes em *cloud computing*, uma vez que diversas das principais ameaças já estão mapeadas, o que auxiliará nosso trabalho visto que testes de penetração estão previstos como trabalhos futuros.

O estudo de [Albaroodi et al. 2014] apresenta desafios de segurança existentes nos três modelos de serviço (SaaS, PaaS e IaaS). Porém, o foco na análise consiste na camada de IaaS, especificamente na ferramenta OpenStack, que por sua vez, tem a função de gerenciar a infraestrutura virtual em ambientes de nuvem. A análise feita por [Albaroodi et al. 2014] tem uma metodologia parecida com a que é feita neste presente estudo, pois é analisado como a ferramenta trata sobre alguns aspectos de segurança, ou quando é o caso, tecnologias que estão incluídas na mesma para prover segurança no ambiente de IaaS. Os resultados são interessantes, pois a análise encontrou algumas vulnerabilidades consideradas graves na ferramenta OpenStack, como não possuir critérios mínimos para a criação de senhas, resulta em senhas fracas. Outra questão é o armazenamento das mesmas serem em forma de texto. Ainda, a comunicação entre os componente da ferramenta não possuem nenhum tipo de criptografia. Baseando-se que a ferramenta OpenStack é bem fragmentada e todos os componentes são praticamente APIs distintas, toda a comunicação interna da ferramenta pode ser facilmente burlada.

Já no estudo de [Dukaric and Juric 2013] é proposto uma taxonomia unificada para ferramentas de IaaS, para uma possível comparação entre elas. Essa taxonomia é baseada em sete camadas necessárias para que uma ferramenta de IaaS seja completa. A taxonomia por si é um resultado interessante, mas em algumas camadas ela não é tão abrangente. O estudo também apresenta um *survey* comparando sete ferramentas, sendo elas quatro de código aberto e três comerciais de código fechado através do *framework* criado. Com essa sumarização de informações é possível verificar que ferramentas de código aberto são consideravelmente mais pobres que as de código fechado, principalmente em aspectos da camada de segurança, gerenciamento e camada de controle. De certa forma, o presente estudo se baseia na metodologia de [Dukaric and Juric 2013], especificamente utilizando a taxonomia proposta da camada de segurança, estendendo com aspectos que não foram levados em conta.

Como evidenciado na Tabela 1, este estudo apresenta semelhanças com os demais estudos e também diferenças em outros aspectos. Referente ao método de pesquisa, este trabalho se utiliza de todos, começando por uma avaliação quantitativa para encontrar o número de mecanismos de segurança. Assim, foi possível traçar um comparativo entre as ferramentas de forma qualitativa. Quanto ao desenvolvimento, o foco deste artigo foi um levantamento dos mecanismos presentes nas ferramentas de IaaS seguidos de uma análise baseada na taxonomia proposta por [Dukaric and Juric 2013]. Diferente dos outros trabalhos, os resultados alcançados vão em direção a uma análise comparativa, discutindo a segurança presente através dos mecanismos oferecidos pelas ferramentas. Isso possibilita uma visão mais detalhada sobre a segurança para auxiliar na tomada de decisão. Além

**Tabela 2. Comparação da Camada de Segurança das ferramentas**

| *Item* | *OpenStack* | *OpenNebula* | *CloudStack* |
|---|---|---|---|
| **Acesso Seguro** | Secure HTTPS Proxy Certificate Authority (CA) | Secure HTTPS Proxy Certificate Authority (CA) | Secure HTTPS Proxy Certificate Authority (CA) |
| Autenticação | Keytone (usuário/senha, Token e LDAP) | Usuário/Senha, SSH, X509, LDAP | Usuário/Senha, LDAP |
| Autorização | Keystone (LDAP, OAuth, Open ID e SAML) | Auth Subsystem 3.0 | Interno (SAML 2.0 Plugin) e LDAP |
| Grupos de Segurança (Isolamento de Redes) | Interno (ACL) | Interno (ACL) | Interno (ACL) |
| *Single Sing-on* | Externo (SAML (Shibboleth ou Mellon)) com o OpenID Connect | / | Interno (SAML 2.0 Plugin) |
| Monitoramento de Segurança | / | / | / |
| **Controle de Permissões Avançadas** | / | / | Possui |

disso, foi possível apontar e sugerir aspectos não considerados na taxonomia.

## 4. Comparação

Essa seção apresenta a avaliação comparativa da camada de segurança das ferramentas OpenStack, OpenNebula e CloudStack. A infraestrutura como serviço é um termo bem amplo, e diversas tecnologias trabalham juntas para que seja possível entregar CPU, memória, rede e armazenamento aos usuários e administradores de IaaS. No contexto que IaaS é a base das demais camadas (PaaS e SaaS), a segurança dela é muito importante para a nuvem como um todo.

Para avaliação das ferramentas, elas foram implantadas em ambientes homogêneos, comparando os itens da camada de segurança proposta por [Dukaric and Juric 2013]. Esta avaliação, baseou-se na documentação oficial bem como em experimentos testados em seus respectivos ambientes. A versão da ferramenta OpenStack utilizada na comparação é a KILO, na OpenNebula a versão 4.12 e CloudStack 4.5.2.

Para evidenciar a comparação, são levantadas características de cada ferramenta perante os itens que abordam a camada de segurança. Essas informações foram tabuladas para facilitar a visualização e entendimento, onde o carácter "/" representa que a ferramenta não possui determinado recurso e nas linhas com preenchimento cinza e negrito é evidenciado itens que foram adicionalmente elencados perante a taxonomia. A comparação da camada de segurança é apresentada na Tabela 2. Os itens adicionalmente elencados em relação a [Dukaric and Juric 2013] na camada de segurança, foram cuidadosamente selecionados para não citar itens que pertencem à outras camadas da taxonomia.

Através do estudo realizado sobre a camada de segurança das ferramentas, é visto que a taxonomia não abrange a segurança no acesso do usuário ou administrador à *dashboard* das ferramentas, e como as ferramentas tratam o acesso seguro às instâncias. Ainda, é elencado o item Controle de Permissões Avançadas (conceder ou remover acesso a recursos específicos no ambiente). A comparação das ferramentas tem um viés para o administrador de IaaS, estes que efetivamente gerenciam os recursos de IaaS no ambiente de nuvem. Por outro lado, os usuários são analistas ou gerentes de TI, pois eles que definem qual é a infraestrutura ideal para o correto funcionamento de suas plataformas ou aplicações.

Através da comparação das três ferramentas, é visto na Tabela 2 que elas possuem recursos nos itens que foram adicionados. O Acesso Seguro refere-se a qual mecanismo de segurança é utilizado pelas ferramentas para garantir a segurança na conexão via web entre usuário ou administrador de IaaS e a nuvem, esse item refere-se também ao acesso seguro do usuário ou administrador às instâncias da nuvem. Nesse item em questão, as três ferramenta OpenStack, OpenNebula e CloudStack utilizam "*Secure HTTPS Proxy Certificate Authority*", que é uma camada adicional de segurança implementada sobre o HTTP, onde é verificado a autenticidade do navegador do usuário ou administrador de IaaS e do servidor por meio de certificados de autoridade. Esses se baseiam em chaves públicas e privadas. Conforme [Rescorla and Schiffman 1999], o HTTPS *(Hyper Text Transfer Protocol Secure)* é uma forma segura para troca de informações, seja na rede local, ou pela internet.

Ressalta-se que de acordo com a visão do administrador, o acesso seguro pode acontecer de diferentes maneiras. O mecanismo é mais efetivo em duas circunstancias, no acesso à *dashboard* (interface gráfica web) e no acesso remoto via VNC. Em ambos os casos, a comunicação ocorre através do estabelecimento de um canal seguro sobre HTTP com SSL/TLS. Em todas as ferramentas avaliadas, diferentes algoritmos de criptografia podem ser utilizados para a geração das chaves, ficando a critério do administrador escolher a complexidade necessária, como por exemplo, o tamanho da chave criptográfica.

No item autenticação, a ferramenta OpenStack utliza o componente Keystone (usuário/senha, Token ou LDAP), este que ainda tem a função de efetuar a autenticação entre os serviços da ferramenta. Na ferramenta OpenNebula, estão disponíveis usuário/ senha, SSH, X509 e LDAP. Na CloudStack, é possível autenticar através de LDAP e usuário/senha.

Ainda na autenticação, a ferramenta OpenNebula suporta via SSH. Apesar de não ser um mecanismo de autenticação em si, o SSH implementa um repositório de chaves. Ele é a própria CA (autoridade certificadora), usando inclusive x509 e sendo a sua própria PKCS (padrão de armazenamento de chaves públicas). Mantemos o item na lista, pois ele é uma ferramenta que provê autenticação independentemente dos outros módulos do OpenNebula.

No item autorização, as ferramentas três possuem características diferentes, a OpenStack utiliza o componente Keystone (LDAP, OAuth, OpenID e SAML). A OpenNebula usa seu sistema interno (*Auth Subsystem 3.0*), e na CloudStack, é utilizado o plugin (SAML 2.0) ou LDAP.

Indo mais a fundo na segurança das ferramentas, as redes virtuais são importantes para os usuários da nuvem, pois toda a comunicação do ambiente é feita através delas. Na comparação, foi visto que as 3 ferramentas possuem controles internos através de ACLs (Lista de Controle de Acesso), onde são criados grupos de segurança e cada um deles possuem regras que são baseadas em políticas específicas.

O item *Single Sing-on* é muito importante quando se utiliza o conceito de federação (sistemas distribuídos em nuvens), no entanto, é apenas encontrado nas ferramentas OpenStack (de forma externa com o Shibboleth e OpenID) e CloudStack (de forma nativa através do *plugin SAML 2.0*). A ferramenta OpenNebula ainda não disponibiliza esse recurso.

Já no item de monitoramento de segurança, nenhuma das ferramentas apresenta

qualquer tipo de recurso que monitore ou verifique se há eventos desconhecidos ou aplicações maliciosas em instâncias ou no ambiente de nuvem. Esses recursos avançados são geralmente encontrados em ferramentas de IaaS de código fechado na utilização de nuvens públicas. Um exemplo de provedor com que dispões desses recursos é a *Amazon Web Services* (AWS).

Essas aplicações são importantes em ambientes de nuvem pública, pois milhares de clientes utilizam e compartilham desses ambientes, seja com plataformas ou aplicações, e um software mal intencionado pode gerar um grande impacto negativo ao ambiente (perda de dados, indisponibilidade, vazamento de informações).

O item Controle de Permissões Avançadas é importante, pois o administrador da nuvem consegue liberar para determinados usuários alguns recursos específicos, sem ter que recriar a conta ou mudar o grupo dele no sistema. Na comparação é visto que apenas a ferramenta CloudStack possui esse tipo de controle (liberar acesso a *templates*, VPN (rede privada virtual), VLAN (rede local virtual), utilização de IPs Públicos). Esse controle não é encontrado nas ferramentas OpenStack bem como OpenNebula.

## 5. Discussão dos Resultados

Nesta seção discute-se os resultados de uma maneira mais ampla das três ferramentas (OpenStack, OpenNebula e CloudStack), abordando qual o impacto das características individuais na totalidade das ferramentas e analisando os resultados da Tabela 2, uma vez que ela é baseada na taxonomia de [Dukaric and Juric 2013]. Ainda, são apresentados itens que não foram considerados pela taxonomia mas foram elencados como itens relevantes para a segurança de uma ferramenta de IaaS. O gráfico da Figura 2 representa de forma quantitativa o suporte aos mecanismos encontrados.

O "Acesso Seguro" é disponibilizado pelas três ferramentas, mas de forma diferente, a OpenStack utiliza TLS *camada de transporte segura* para encriptar a conexão VNC com a VM. No estudo de [Dierks 2008], é evidenciado que é difícil garantir que uma conexão seja totalmente segura, mesmo utilizando criptografia no canal de comunicação, pois a segurança da conexão também depende da aplicação do cliente (usuário) e do servidor (ferramenta de IaaS).

No contexto de autenticação, o suporte de tokens/certificado x509 permite à ferramenta uma maior flexibilidade em termos de segurança na autenticação. Ao contrário de um simples usuário e senha armazenados e criptografados em um banco de dados, a autenticação via infraestrutura de chave publica/privada permite um controle mais efetivo, já que as senhas contidas nas chaves são mais longas e ficam armazenadas em uma estrutura específica (PKCS#12).

Considerando o contexto de nuvens privadas, o uso de autorização através de SAML é mais relevante, pois para o usuário ser autorizado, deve haver um acordo explícito de confiança entre os servidores, não sendo aceitas identidades de domínios desconhecidos. O OpenID é muito mais flexível neste sentido, ele permite que qualquer servidor atribua identidades através de requisições HTTP. Desta forma, quando o serviço é aberto e disponibilizado publicamente, o uso do OpenID é o mais recomendado. Ao contrário, quando deve haver o controle de todos os usuários, mesmo em aplicações multidomínio, a recomendação é pelo uso do SAML. Chama a atenção a ausência de autorização através do protocolo OAuth, provavelmente o mais utilizado em aplicações na Internet.

No item grupos de segurança, todas as ferramentas possuem mecanismos para o isolamento de redes e recursos, isso acontece de forma interna em cada ferramenta, através de ACLs (Controle de Lista de Acesso). No estudo de [Dawoud et al. 2010], é evidenciado que o virtualizador é quem oferece os recursos virtuais. Dessa forma, eles são diretamente dependentes do virtualizador, este que tem grande influência na segurança do ambiente a um nível que a ferramenta não possui controle.

O item *Single Sing-on* é suportado em duas ferramentas: Na OpenStack acontece de forma externa. Isso pode ser visto de forma negativa, pois depende da instalação do administrador de IaaS, assim sendo suscetível a erros na implantação. Já a CloudStack, por padrão suporta determinado recurso. Na ferramenta OpenNebula, esse recurso não é encontrado.

É importante ressaltar que a taxonomia apresentada por [Dukaric and Juric 2013], contém um equivoco na identificação dos elementos de segurança. Mecanismos de *Single-sign on* são protocolos de autenticação que estabelecem confiança mútua em servidores federados, permitindo que usuários de múltiplos domínios autentiquem e carreguem suas preferências de identidade sem a necessidade de um novo registro a cada domínio. Desta forma, entende-se que os dois itens da taxonomia se sobrepõe, inclusive sendo gerenciados pelo mesmo módulo do sistema. Um exemplo é a ferramenta OpenStack que faz uso do módulo Keystone para disponibilizar ao usuário serviços de autenticação e *Single-sign on*.

Um ponto negativo das três ferramentas, é que nenhuma possui algum monitoramento de segurança, ou seja, algum recurso proativo que monitore e detecte eventos desconhecidos no ambiente de IaaS. Tal recurso é frequentemente encontrado em ferramentas de código fechado, geralmente em nuvens públicas. Reforça-se que está se considerando o monitoramento "ativo" de características de segurança, com interface para controle administrativo. Ainda que todas as ferramentas possuem um monitoramento "passivo", em baixo nível elas garantem as políticas de controle de acesso, isolamento entre usuários em

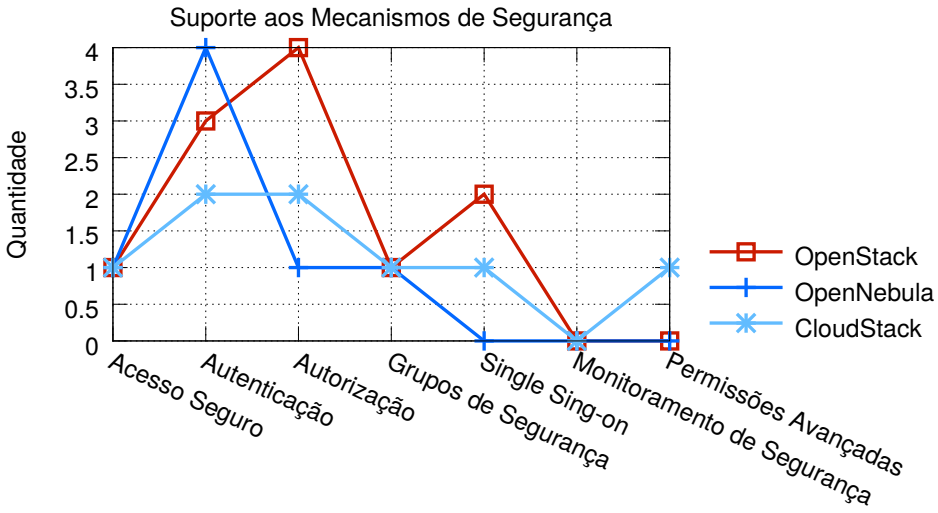


**Figura 2. Avaliação quantitativa dos mecanismos de segurança**

ambiente de processamento e armazenamento, gerando logs em caso de erros ou tentativas de subversão do sistema. Porém, estes dados não estão facilmente disponíveis aos usuários do sistema.

No item Controle de Permissões Avançadas, apenas a ferramenta CloudStack suporta esse recurso, onde é possível conceder permissões específicas a determinados usuários, sem a necessidade de trocá-lo de grupo. Nas ferramentas OpenStack e OpenNebula, não é encontrado essa opção.

Analisando características das ferramentas e ambientes de implantação, se torna relativo a quais cenários cada uma das ferramentas é recomendada, sempre levando em conta que quanto maior e mais complexa a ferramenta, mais pontos de vulnerabilidade podem ser encontrados. Essa relação tamanho versus segurança é fundamental para determinar em qual cenário cada ferramenta se encaixa. Dessa forma, a complexidade e robustez da ferramenta OpenStack reflete em uma capacidade maior de customização, porém, pode trazer dificuldades para a definição de ambientes e configuração, que pode resultar em mais vulnerabilidades.

Para implantação de ambientes complexos de nuvem existem aspectos importantes para uma ferramenta (gerenciamento, segurança, controle de usuários, controle de uso, entre outros). Voltando-se para a segurança, ferramentas mais completas tendem a ser recomendadas para implantações complexas, pois podem mais facilmente suprir demandas e também vão contar com uma disponibilidade maior de administradores e arquitetos. Por outro lado, ferramentas mais simples como OpenNebula são sugeridas para ambiente menores, com menos usuários, e que demandam de menos controles avançados. Ela é facilmente implantada e dificilmente terá o suporte para implantações complexas.

Toda tecnologia que envolva algum tipo de programação pode possuir alguma vulnerabilidade, mesmo que haja correções para esses problemas, estes acarretam em novas vulnerabilidades. Desta forma, seguindo a concepção que quanto maior for o suporte para tecnologias em uma ferramenta, mais potenciais vulnerabilidades ela possui. Do ponto de vista da segurança, a ferramenta OpenStack é extremamente modular e possui um amplo suporte a tecnologias, mas isso pode ser visto como uma vulnerabilidade em potencial, pois não existe criptografia ou verificação das mensagens entre os módulos. Já a ferramenta CloudStack e OpenNebula possuem uma arquitetura mais compactada, podendo ser mais segura nesse aspecto.

É visto que existem diferenças entre as ferramentas, como a arquitetura ou até o suporte na utilização de tecnologias. De certa forma, já ocorre uma segregação de qual ambiente cada ferramenta se adapta melhor, levando em conta o nível de gerenciamento requirido pela ferramenta, bem como os requisitos de instalação e configuração do ambiente. Nesse ponto, as ferramentas CloudStack e OpenNebula podem ser uma melhor opção para pequenas e médias empresas, uma vez que possui uma interface mais simples e amigável, enquanto a OpenStack por suportar uma quantidade maior de tecnologias é mais aconselhada para grandes empresas, que possuem um grande ambiente e necessitam de maior modularidade, permitindo um maior controle sobre integrações e recursos.

## 6. Conclusões e Trabalhos Futuros

A computação em nuvem surgiu como um paradigma inovador oferecendo recursos computacionais na forma de serviço, fazendo com que a computação fosse disponibilizada

como uma utilidade. Porém, com isso surgiram diversas preocupações com a sua utilização, e a mais pertinente de todas é relacionada com segurança da informação e privacidade nesses ambientes. Nesse estudo, três das principais ferramentas de código aberto para a implantação de nuvens IaaS tiveram aspectos no suporte para mecanismos de segurança analisados e comparados. Na comparação da Figura 2, ficou evidente que as ferramentas possuem contrastes. A OpenStack apresentou os níveis mais elevados de suporte a recursos de segurança, enquanto CloudStack teve médias regulares e OpenNebula mostrou-se como uma ferramenta mais simplista.

Como trabalhos futuros espera-se revisar e estender a taxonomia proposta por [Dukaric and Juric 2013] de acordo com a teoria da segurança de sistemas de informação. Uma vez que foi constatada a sobreposição de elementos como *single-sign on* e autorização, bem como a ausência de elementos de controle de privacidade. A taxonomia trata todos os elementos de segurança como uma camada única, porém, os métodos de segurança aplicados em diferentes níveis da arquitetura de *cloud computing* também variam. Para uma melhor granularidade da sua descrição, é necessário observar as suas diferenças. Também temos como objetivo uma análise de segurança mais profunda, verificando vulnerabilidades através de *pentesting* (Teste de penetração).

## Agradecimentos

Esta pesquisa foi realizada com o apoio do projeto HiPerfCloud[2]. Os autores agradecem o suporte financeiro da Abase Sistemas[3] e da Sociedade Educacional Três de Maio (SETREM)[4].

## Referências

[Albaroodi et al. 2014] Albaroodi, H., Manickam, S., and Singh, P. (2014). Critical review of openstack security: Issues and weaknesses. *Journal of Computer Science*, pages 23–33.

[Bishop 2005] Bishop, M. (2005). *Introduction to computer security*. Addison-Wesley Boston, MA.

[Buyya et al. 2013] Buyya, R., Vecchiola, C., and Selvi, S. (2013). *Mastering Cloud Computing: Foundations and Applications Programming*. Mastering Cloud Computing: Foundations and Applications Programming. Elsevier Science.

[CloudStack 2015] CloudStack (2015). CloudStack (Official Page) <https://cloudstack.apache.org/>. Last access in October, 2015.

[Dawoud et al. 2010] Dawoud, W., Takouna, I., and Meinel, C. (2010). Infrastructure as a service security: Challenges and solutions. In *Informatics and Systems (INFOS), 2010 The 7th International Conference on*, pages 1–8. IEEE.

[Dierks 2008] Dierks, T. (2008). The transport layer security (tls) protocol version 1.2.

[Dukaric and Juric 2013] Dukaric, R. and Juric, M. B. (2013). Towards a Unified Taxonomy and Architecture of Cloud Frameworks. *Future Gener. Comput. Syst.*, 29(5):1196–1210.

[2]http://hiperfcloud.setrem.com.br/

[3]http://www.abase.com.br/

[4]http://www.setrem.com.br/

[ISO JTC1/SC38 Technical Report 2014] ISO JTC1/SC38 Technical Report (2014). ISO/IEC 17789:2014. Information Technology - Cloud Computing - Reference Architecture. last access in Sept, 2015.

[Maron et al. 2014] Maron, C. A. F., Griebler, D., and Schepke, C. (2014). Comparação das Ferramentas OpenNebula e OpenStack em Nuvem Composta de Estações de Trabalho. In *14th Escola Regional de Alto Desempenho do Estado do Rio Grande do Sul (ERAD/RS)*, pages 173–176, Alegrete, RS, Brazil. Sociedade Brasileira de Computação.

[OpenNebula 2015] OpenNebula (2015). OpenNebula (Official Page) <http://opennebula.org/>. Last access in October, 2015.

[OpenStack 2015] OpenStack (2015). OpenStack roadmap <http://openstack.org/>. Last access October, 2015.

[Rescorla and Schiffman 1999] Rescorla, E. and Schiffman, A. (1999). The secure hypertext transfer protocol.

[Roveda et al. 2015a] Roveda, D., Vogel, A., and Griebler, D. (2015a). Understanding, Discussing and Analyzing the OpenNebula and OpenStack's IaaS Management Layers. *Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação*, 3(1):15.

[Roveda et al. 2015b] Roveda, D., Vogel, A., Maron, C. A. F., Griebler, D., and Schepke, C. (2015b). Analisando a Camada de Gerenciamento das Ferramentas CloudStack e OpenStack para Nuvens Privadas. In *13th Escola Regional de Redes de Computadores (ERRC)*, Passo Fundo, Brazil. Sociedade Brasileira de Computação.

[Shroff 2010] Shroff, G. (2010). *Enterprise Cloud Computing: Technology, Architecture, Applications*. Cambridge University Press.

[Thome et al. 2013] Thome, B., Hentges, E., and Griebler, D. (2013). Computação em Nuvem: Análise Comparativa de Ferramentas Open Source para IaaS. In *11th Escola Regional de Redes de Computadores (ERRC)*, page 4, Porto Alegre, RS, Brazil. Sociedade Brasileira de Computação.

[Vaquero et al. ] Vaquero, L. M., Rodero-Merino, L., and Morán, D. Locking the sky: a survey on iaas cloud security. *Computing*.

[Vogel et al. 2016] Vogel, A., Griebler, D., Maron, C. A. F., Schepke, C., and Fernandes, L. G. L. (2016). Private IaaS Clouds: A Comparative Analysis of OpenNebula, CloudStack and OpenStack. In *24rd Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, Heraklion Crete, Greece. IEEE.

[Zhou et al. 2010] Zhou, M., Zhang, R., Xie, W., Qian, W., and Zhou, A. (2010). Security and privacy in cloud computing: A survey. In *Semantics Knowledge and Grid (SKG), 2010 Sixth International Conference on*, pages 105–112. IEEE.