Capítulo 2

Aspectos de segurança e privacidade em ambientes de Computação em Nuvem

Arlindo Marcon Jr[1,2], Marcos Laureano[1,2], Altair Santin[1], Carlos Maziero[1]

[1]Programa de Pós-Graduação em Informática – PPGIa
Pontifícia Universidade Católica do Paraná – PUCPR

[2]Instituto Federal de Educação, Ciência e Tecnologia do Paraná – IFPR

Abstract

Cloud computing aims to provide on-demand access to a pool of computing resources, in a dynamic and easily scalable environment. The use of services provided by third parties allows to minimize efforts in local information technology management, and gives benefits in mobility, scalability, and availability. This short course intends to explore the state of the art in the security and privacy areas of cloud computing environments. Initially, the fundamental aspects of cloud computing will be presented. After a short review of the main concepts in security and privacy, it presents a deeper analysis of the relevant risks and threats in cloud environments, as well as the most known approaches to deter them. The text also discusses some open problems and tentative solutions proposed in the litterature.

Resumo

A computação em nuvem visa prover acesso sob demanda a um pool *de recursos computacionais em um ambiente dinâmico e facilmente escalável. A partir de serviços prestados por terceiros se minimizam as preocupações de gerenciamento de tecnologia de informação local, trazendo vantagens de mobilidade, escalabilidade e disponibilidade. Este minicurso visa explorar o estado da arte nas áreas de segurança e privacidade no contexto de computação em nuvem. Inicialmente serão apresentados os aspectos fundamentais de computação em nuvem. Após uma revisão dos principais conceitos de segurança e privacidade, serão discutidos em maior profundidade os riscos e ameaças relevantes nos ambientes de nuvem, bem como as abordagens mais conhecidas para mitigá-los. Ao longo do texto serão discutidos problemas em aberto e tentativas de solução propostas na literatura.*

2.1. Introdução

Computação em nuvem (*Cloud Computing*) visa prover acesso sob-demanda para um *pool* de recursos computacionais (e.g. rede, armazenamento, serviços). Estes recursos podem ser rapidamente providos/liberados com pouco esforço de gerenciamento, pois o ambiente é nativamente dinâmico e facilmente escalável [Mell e Grance 2009]. A principal motivação para computação em nuvem é que a partir de serviços prestados por terceiros se eliminam as preocupações de gerenciamento de tecnologia de informação local, i.e., instalação, configuração e atualização de sistemas, e manutenção da infraestrutura computacional física [Hayes 2008]. Ou seja, a computação em nuvem oferece vantagens relacionadas a mobilidade, escalabilidade, disponibilidade, e implantação de sistemas computacionais.

A definição mais aceita para descrever a computação em nuvem é composta de sete características fundamentais: três modelos de serviço e quatro abordagens de implantação [Mell e Grance 2009]. Os modelos de serviço são compostos pela (a) *Infraestrutura como Serviço*/IaaS – fornece recursos computacionais como processamento, armazenamento, rede etc., (b) *Plataforma como Serviço*/PaaS – permite utilizar a infraestrutura de nuvem para criar e implantar novas aplicações próprias ou para prover suporte para nível de SaaS e (c) *Software como Serviço*/SaaS – provê aplicações à nuvem para serem consumidas sob-demanda. A implantação dos modelos podem seguir uma abordagem (i) *pública* – com acesso disponibilizado para o público em geral, (ii) *privada* – de uso exclusivo de uma organização, (iii) *comunitária* – compartilhada por organizações com interesses comuns ou (iv) *híbrida* – qualquer tipo de combinação entre as categorias anteriores.

Para a infraestrutura o mecanismo de virtualização é uma das principais abordagens. Este permite a flexibilização do uso da camada de hardware. As máquinas virtuais proveem ambientes de processamento independentes e isolados, podendo ser instanciadas e destruídas sob demanda. Dessa forma, o ambiente de máquinas virtuais constitui uma base bastante adequada para a construção de infraestruturas de computação em nuvem [Grobauer et al. 2010].

O modelo de serviço, operacional e as tecnologias utilizadas para prover os serviços do ambiente de computação em nuvem apresentam diferentes níveis de riscos se comparado ao ambiente tradicional de tecnologia de informação [CSA 2009]. O provimento de recursos sob demanda para o processamento e armazenamento massivo de dados está sujeito a falhas de segurança, abusos com relação à privacidade, violação de direitos autorais, etc. Preocupações este aspectos de segurança computacional estão impedindo a ampla adoção da computação em nuvem.

Este minicurso tem como objetivo explorar o estado da arte nas áreas de segurança e privacidade no contexto de computação em nuvem (*cloud computing*). Inicialmente serão apresentados os aspectos fundamentais de computação em nuvem. Após uma breve revisão dos principais conceitos de segurança e privacidade, serão discutidos em maior profundidade os riscos e ameaças relevantes nos ambientes de nuvem, bem como as abordagens conhecidas para mitigá-los. Ao longo do texto serão discutidos problemas em aberto e tentativas de solução propostas na literatura.

2.2. Computação em nuvem

2.2.1. Introdução

Há quase 50 anos foi criado o sistema de compartilhamento de tempo – *time-sharing*. Este sistema fornecia acesso a poder computacional para usuários que não possuíam seu próprio *mainframe*, usando sistemas do tipo *hub-and-spoke*. No começo dos anos 1980 com a chegada dos computadores pessoais, programas e dados não dependiam mais de um centro computacional para executar. Cada indivíduo passou a controlar seu próprio ambiente de trabalho, customizando-o de acordo com suas necessidades. O modelo cliente-servidor, introduzido na mesma época, oferecia um serviço que podia ser invocado através da rede para atender uma necessidade do cliente [Bhattacharjee 2009].

Atualmente, softwares de prateleira ainda dominam o mercado, porém este paradigma está mudando para os ambientes de computação em nuvem – *Cloud Computing*. Tarefas computacionais podem ser migradas dos computadores de mesa e servidores corporativos para a nuvem computacional [Erickson et al. 2009]. A mudança de paradigma marca a inversão de uma tendência que perdurou por muitos anos, afetando todos os níveis do ecossistema computacional, incluindo usuários, desenvolvedores, gerentes de Tecnologia da Informação e os fabricantes de hardware [Hayes 2008].

A nuvem computacional pode ser utilizada para hospedar softwares em centros computacionais disponíveis e acessíveis via Internet. Na topologia atual não existe um ponto central – *hub*, um terminal cliente pode se comunicar com muitos servidores ao mesmo tempo, sendo que estes podem estar trocando informações entre si. A nuvem computacional pode ter uma ou mais centrais de gerenciamento – formada por vários provedores administrando diferentes domínios. Novas funcionalidades para aplicações e serviços podem ser disseminadas a partir de uma central administrativa, sem que o consumidor tenha que se preocupar com a complexidade de gerenciamento do ambiente.

As principais entidades que fazem parte do modelo de computação em nuvem, fornecendo ou interagindo com os serviços podem ser brevemente descritas como: (i) *provedor* – entidade que gerencia e fornece os serviços hospedados nas infraestruturas físicas. O provedor tem controle sobre os recursos computacionais – processador, memória etc.; (ii) *consumidor* – empresa ou organização que contrata e utiliza os serviços de um provedor; (iii) *usuário* – entidade que pode estar vinculada a um consumidor ou agindo por conta própria – o usuário final do serviço.

A migração de sistemas tradicionais para os serviços fornecidos pela nuvem pretende reduzir os custos de manutenção da infraestrutura de TI (Tecnologia da Informação)do consumidor, oferecendo as seguintes vantagens [Zhang et al. 2010]: economia em servidores, armazenamento, rede, licenças de software, energia, resfriamento e bens materiais; redução de trabalho na administração de sistemas; redução do tempo de configuração; diminuição de equipes de trabalho; desenvolvimento de aplicações com ciclo de vida mais curto e consequente redução do tempo de disponibilização de novos produtos e serviços no mercado; maior confiabilidade com custos menores e redução de gastos com manutenção, redução de custos com atualizações de hardware/infraestrutura.

A computação em nuvem também oferece vantagens relacionadas a mobilidade, escalabilidade e disponibilidade de sistemas. A escalabilidade é uma das maiores vantagens

para quem pretende implantar um serviço usando o modelo de computação em nuvem, pois é possível demandar recursos adicionais mesmo se o número de usuários crescer de forma imprevisível. Empresas consumidoras que necessitam de serviços de TI estão considerando a possibilidade de utilizar provedores de serviços terceirizados (*off-premise*), tirando proveito das vantagens oferecidas pela computação em nuvem.

Para facilitar a entrada de novas empresas no mercado, a computação em nuvem aplica o modelo de negócio *pay-as-you-go*. Ou seja, a cobrança é feita de acordo com a utilização de recursos e serviços. Na computação em nuvem, os gastos com capital são convertidos em gastos operacionais [Armbrust et al. 2010]. Os serviços fornecidos pelo provedor de computação em nuvem podem ser distribuídos e utilizados de maneira não uniforme, de acordo com a necessidade do consumidor. Na comunidade de rede há uma situação análoga, onde a largura de banda tem preço baseado em uso (*usage-based pricing*). A computação em nuvem é utilizada pelos consumidores e contabilizada pelo provedor para que o uso de recursos e serviços possa ser faturado [Bhattacharjee 2009].

A computação em nuvem permite que os consumidores utilizem a quantidade de recursos necessários para realizar testes com novos sistemas. Se um projeto falhar durante sua fase inicial, por exemplo, o consumidor do serviço da nuvem investiu pouco no negócio, podendo facilmente alterar seu tipo de negócio ou seus provedores de serviço. No modelo tradicional (*on-premise*) o contratante precisa gastar previamente em licenças, infraestrutura, consultoria etc.; se o projeto falhar esse investimento prévio vira prejuízo.

Em grandes *data centers* as máquinas físicas estão totalmente utilizadas em torno de 20% a 30% do tempo. Com a aplicação da computação em nuvem e das respectivas tecnologias de suporte (e.g. virtualização), a utilização dos recursos é maximizada, reduzindo o tempo ocioso de cada máquina. A camada de virtualização abstrai o hardware, intermediando o acesso de várias aplicações aos mesmos recursos físicos: processador, memória etc. Esta tecnologia permite realizar a consolidação (agrupamento) de cargas de trabalho em poucos servidores físicos, reduzindo os gastos com energia e resfriamento e consequentemente levando a economia em hardware, manutenção etc. A utilização adequada do ambiente de computação em nuvem agrega valor ao negócio de seus consumidores [Bhattacharjee 2009].

A técnica da virtualização permite que sistemas hospedados em um servidor físico sejam transferidos para outros servidores, executando o balanceamento de carga ou cópias de segurança dos sistemas. Com está abordagem, a execução ou restauração de cópias de segurança é concluída em uma pequena fração do tempo que levaria com os servidores físicos tradicionais. No caso de falhas na aplicação ou serviço, uma instância de *backup* (*hot backup*) pode assumir imediatamente o lugar da faltosa. Neste caso, a interrupção no serviço pode passar despercebida para os usuários. As garantias fornecidas pelo provedor para seus consumidores podem ser definidas através de contratos em nível de serviço – SLAs (*Service Level Agreements*) [Kandukuri et al. 2009].

A maioria dos provedores de nuvem oferecem garantias de tempo de atividade (*uptime*) em seus SLAs. Um dos problemas relacionados ao SLA é a dificuldade para expressar e implementar o contrato em nível computacional – e.g. como fornecer garantias de tempo de atividade para uma transação se esta envolve um fluxo de dados através da Internet. As empresas tradicionais ainda possuem dúvidas com relação a transferência de

seus dados internos para fora dos limites de seu filtro de pacotes (*firewall*). Certos países têm suas próprias regras quanto ao lugar onde as empresas podem armazenar seus dados, por exemplo, Canadá e Estados Unidos. Para tentar contornar essas barreiras, provedores de nuvem estão implantando *data centers* em várias partes do mundo.

Empresas que optarem por utilizar um determinado provedor de computação em nuvem podem basear suas escolhas em vários fatores: preço, confiabilidade, disponibilidade, abrangência, suporte etc. Mesmo que um único provedor atenda estes requisitos, outros provedores podem ser utilizados, fornecendo diferentes conjuntos de funcionalidades, como por exemplo: armazenamento da *Amazon*, poder computacional do *Google* e CRM (*Customer Relationship Management*) da *Salesforce*. Tudo isto pode ser feito através de APIs de acesso padronizadas, permitindo que as aplicações permaneçam inalteradas, sendo necessário alterar apenas os provedores de computação em nuvem.

Apesar da existência de grandes provedores de computação em nuvem (e.g. *Amazon*, *Google*, *Saleforce*) existem algumas empresas menores participando deste mercado (e.g. *RightScale*, *Hyperic*, *Mosso*, *Elastra*). Porém, algumas destas companhias pode não sobreviver à concorrência, abandonando o mercado ou sendo adquiridas por outras empresas. Neste caso, os dados e aplicações de seus consumidores e usuários precisam ser transferidos para outro provedor, ou retornar para dentro dos limites organizacionais do consumidor. A padronização neste caso é fundamental, fornecendo um nível de garantia adicional aos consumidores – além dos contratos em nível de serviço.

O tráfego de dados entre a nuvem computacional e o consumidor ou entre serviços hospedados em diferentes provedores gera latência, que fica ainda maior se comparada com o acesso local ao *data center* de uma empresa tradicional. A tecnologia para transferências de dados que muitas nuvens computacionais disponibilizam é baseada nos métodos *get/post* do protocolo HTTP. Porém, este protocolo não foi projetado para atender este tipo de demanda. Para auxiliar na resolução deste problema, a conexão entre o consumidor e o provedor deve ser geograficamente a mais próxima possível. Abordagem esta que é contraditória a proposta da computação em nuvem – fornecer serviços de maneira transparente à localização.

No caso de um *data center* corporativo tradicional ser atacado, o efeito do dano será sentido somente neste domínio, ou por algumas entidades que tenham negócios com esta organização. Porém, se um grande domínio for atacado (e.g. *Amazon*, *Google*) o efeito será percebido por todos, possivelmente nas interfaces dos softwares dos consumidores. Adicionalmente, estes hospedam grandes grupos de consumidores e possivelmente os sistemas de empresas inteiras. A ocorrência de alguma falha pode ter um grande impacto nos negócios de várias entidades. Alguns provedores de nuvem tentam minimizar estes problemas replicando seus *data centers* – dados e aplicações de seus usuários. Assim, a falha em um *data center* não deverá paralisar todos os sistemas, mas o desempenho de algumas aplicações poderá ser afetado.

O uso da computação em nuvem também pode ser percebido em meios científicos e acadêmicos, sendo geralmente implantada em centros de computação de alto desempenho, *High Performance Computing* – HPC [Ogrizovic et al. 2010]. HPCs acadêmicos estão expandindo suas infraestruturas através da virtualização dos *clusters* locais. Esta abordagem permite o fornecimento de ambientes personalizados para uma grande variedade de

consumidores, atendendo as necessidades específicas de cada tipo de demanda.

De maneira geral, o modelo de computação em nuvem visa prover acesso sob demanda a diferentes camadas da plataforma computacional (e.g. rede, servidores, armazenamento, aplicações, serviços etc.). Estas funcionalidades podem ser rapidamente fornecidas ou liberadas com pouco esforço de gerenciamento ou interação humana. A nuvem visa fornecer alta disponibilidade e elasticidade, sendo composta de cinco características essenciais [Mell e Grance 2009]:

1. *Auto-Atendimento*: o consumidor configura cada recurso computacional conforme sua necessidade, sem exigir interação humana com os provedores de serviço.
2. *Amplo acesso à rede*: os recursos são disponibilizados na rede e acessados através de mecanismos padronizados. Isto possibilita o uso em diferentes plataformas (e.g. celulares, notebooks etc.).
3. *Pool de recursos*: os recursos computacionais do provedor são agrupados. Isto permite servir múltiplos consumidores em um modelo multi-inquilino (*multi-tenant*). Ou seja, os recursos físicos e virtuais são distribuídos e ou redistribuídos dinamicamente de acordo com a demanda do consumidor.
4. *Elasticidade*: os recursos podem ser fornecidos rapidamente e em alguns casos automaticamente. A quantidade de recursos disponibilizados passa para o consumidor a impressão de que a nuvem possui uma infraestrutura ilimitada.
5. *Medição no uso dos serviços*: a nuvem controla e otimiza o uso de recursos fornecendo métricas de acordo com o tipo de serviço sendo fornecido. Tanto o provedor quanto o consumidor podem monitorar e controlar a utilização dos recursos.

2.2.2. Serviços, plataformas e infraestrutura

Ambientes de computação em nuvem são similares a sistemas distribuídos que executam processamento de dados. Os critérios utilizados para definir as diferentes abordagens de nuvens podem ser [Rimal et al. 2009]: a) *arquitetura da nuvem* – serviços disponíveis na Internet através de um ponto de entrada que fornece acesso aos recursos computacionais para implementá-los, b) *virtualização* – tecnologia que abstrai o acoplamento entre o sistema e o hardware, c) *serviços* – implementados por provedores como: *SalesForce*, *Microsoft Azure*, *Amazon* etc., d) *tolerância a faltas* – técnicas adotadas para fornecer serviços confiáveis, e) *segurança* – proteção dos dados processados e armazenados, f) *balanceamento de carga* – redistribuição de carga de trabalho ou redirecionamento das solicitações de acesso, g) *interoperabilidade* – definição de interfaces para permitir a portabilidade de aplicações entre nuvens, h) *armazenamento escalável de dados* – transferência de dados para a nuvem sem preocupação com a forma de armazenamento ou cópias de segurança, i) *escalabilidade horizontal/vertical*: a escalabilidade horizontal é denotada pelo que a nuvem fornece através do balanceamento de carga e a escalabilidade vertical esta relacionada a quantidade de recursos utilizados.

Basicamente os provedores de computação em nuvem podem ser classificados de acordo com o tipo de serviço oferecido e o respectivo modelo de implantação [Mell e Grance 2009]. Os modelos de serviço são (Figura 2.1):

1. *Software como um Serviço* (SaaS): oferece o produto final de computação em nuvem, o software que o consumidor usa. O consumidor não gerencia ou controla a infraestrutura – rede, servidores, sistema operacional, armazenamento e funcionalidades de aplicações. O consumidor usa as aplicações sendo executadas na infraestrutura da nuvem. As aplicações podem ser acessadas através de um *thin client* – navegador web – por exemplo.

2. *Plataforma como um Serviço* (PaaS): oferece recursos para o consumidor implantar na infraestrutura da nuvem suas próprias aplicações, desde que utilizem linguagens de programação e ferramentas suportadas pelo provedor. Nesta abordagem, o consumidor não gerencia ou controla a infraestrutura subjacente – rede, servidores, armazenamento – mas somente controla sua própria aplicação e o sistema que está hospedando as configurações do ambiente.

3. *Infraestrutura como um Serviço* (IaaS): fornece recursos computacionais básicos como processamento, armazenamento, rede etc. Com estes recursos o consumidor pode implantar e executar uma grande variedade de programas – sistemas operacionais e seus aplicativos. Neste nível o consumidor não controla ou gerencia a infraestrutura física da nuvem, mas tem controle limitado sobre componentes da rede, como por exemplo, filtros de pacotes.

- *Identidade para a Nuvem como um Serviço* (IDaaS): o *Cloud Security Alliance* considera o IDaaS um serviço de gerenciamento de identidades para a nuvem, sendo externo as aplicações e aos provedores que utilizam as identidades [CSA 2010a]. O IDaaS é um serviço que fornece gerenciamento de identidade e do ciclo de vida dos usuários, funções de controle de acesso, *Single Sign-On* etc. Este serviço pode ser utilizado pelos modelos SaaS, PaaS e IaaS.

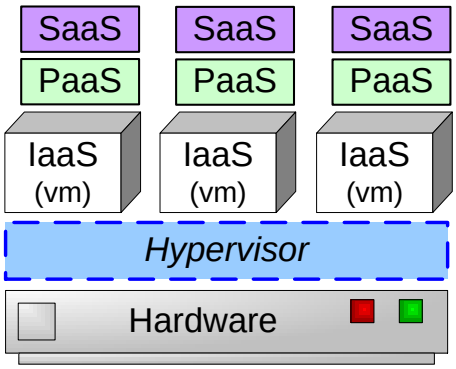

Figura 2.1. Modelos de Serviço.

A computação em nuvem pode ser vista como uma pilha de serviços. Cada camada da pilha oferece serviços construídos com base nas camadas inferiores. Alguns serviços básicos, como medição, contabilização e gerenciamento, se espalham pelas várias camadas, pois são necessários para todos os tipos de serviços oferecidos na nuvem.

A camada IaaS abrange toda a pilha de recursos da infraestrutura – desde as instalações físicas até as plataformas de hardware disponíveis. Essa camada incorpora a

funcionalidade de abstração de recursos, possibilitando a conectividade entre a infraestrutura física e a lógica. A camada PaaS – posicionada acima do IaaS – adiciona um nível de integração com *frameworks* de desenvolvimento de aplicações e funcionalidades de *middleware*. Essa camada provê funções como banco de dados e recursos para troca e enfileiramento de mensagens, permitindo aos desenvolvedores a construção de aplicações sobre o PaaS. As linguagens de programação e as ferramentas utilizadas precisam ser suportadas pelas camadas subjacentes. A camada SaaS – posicionada acima do IaaS e PaaS – provê um ambiente de operação autocontido, utilizado para disponibilizar diferentes serviços para seus consumidores, por exemplo, conteúdos, aplicações, funcionalidades de gerenciamento etc.

Os quatro principais modelos de implantação que podem ser aplicados a computação em nuvem são [Mell e Grance 2009]:

1. *Nuvem privada*: a infraestrutura é operada exclusivamente para atender as necessidades de uma organização, sendo que essa pode ser gerenciada pela organização ou por um terceiro, e sua implementação pode ser local ou remota.

2. *Nuvem baseada em comunidade*: compartilhada por várias organizações que possuem interesses comuns – requisitos de segurança, políticas etc. Esta pode ser gerenciada pelas organizações participantes da comunidade ou por um terceiro, em implementação local ou remota.

3. *Nuvem pública*: a infraestrutura é disponibilizada para o público em geral, podendo pertencer a alguma organização que vende serviços de computação.

4. *Nuvem híbrida*: é uma composição entre dois ou mais modelos de nuvens, por exemplo, privado e público. Estes permanecem como entidades únicas, porém, são ligados por alguma tecnologia especifica – padronizada e aberta ou proprietária. Uma composição de nuvem híbrida pode viabilizar o balanceamento de carga, ou seja, quando a parte privada não consegue mais atender a demanda a parte pública pode fazer esta tarefa – se os dados não forem sensíveis.

Os provedores oferecem seus serviços de forma distinta, diferenciando-se na forma como o consumidor pode obter e como este é capaz de acessar os serviços [Zhang et al. 2010]. A *Amazon* oferece instâncias de máquinas virtuais – que podem ser *Linux*, *Solaris* ou *Windows* – sendo o consumidor livre para instalar suas próprias aplicações. Serviços de armazenamento, banco de dados e gerenciamento de conteúdo também são disponibilizados pela *Amazon*. O *Google* disponibiliza algumas funcionalidades de seu sistema através de uma interface baseada em *Python*, não fornecendo acesso a imagens de um sistema operacional ou qualquer tipo de banco de dados padrão. A *Google* fornece APIs para que os consumidores escrevam aplicações e acessem serviços como correio eletrônico, armazenamento de dados proprietário etc. O provedor *Force.com* é comparável ao *Google* no modelo de negócio, permitindo que o desenvolvedor construa uma aplicação usando um mecanismo de *workflow* e deixando a implementação subjacente da aplicação para a nuvem computacional.

A Arquitetura Orientada a Serviço juntamente com as tecnologias de virtualização podem ser utilizadas para criar um modelo de computação em nuvem reutilizável e configurável – uma arquitetura aberta para nuvens computacionais (*Cloud Computing Open Architecture* – CCOA) [Zhang e Zhou 2009].

Os principais aspectos a serem considerados na definição de uma arquitetura para computação em nuvem podem ser brevemente descritos como: criação de um projeto que possa ser reutilizado e que aplique plataformas de configuração escaláveis; utilização da orientação a serviço e da virtualização para agregar valores práticos e de negócios para as aplicações, sistemas e processos de negócios; modelagem de um conjunto de serviços que seja comuns para as plataformas de nuvens; maximização do valor dos negócios – aplicação de infraestruturas de tecnologia e sistemas de gerenciamento extensíveis.

Para a infraestrutura o mecanismo de virtualização é uma das principais abordagens, por permitir a flexibilização do uso da camada de hardware [Laureano e Maziero 2008]. As máquinas virtuais proveem ambientes de processamento independentes e isolados, podendo ser instanciadas e destruídas sob demanda. Dessa forma, o ambiente de máquinas virtuais constitui uma base bastante adequada para a construção de infraestruturas de computação em nuvem [Grobauer et al. 2010].

2.2.3. Ambientes de computação em nuvem

O consumidor que necessitar de uma grande quantidade de recursos computacionais poderia ter que conectar-se a vários provedores de recursos diferentes para poder satisfazer sua demanda. Assim, o *pool* de recursos fornecido pode ser bastante heterogêneo, tornando a tarefa de utilização complexa, pois envolve o gerenciamento de diferentes contratos, políticas, interfaces, usuários etc. A maioria dos consumidores preferem um ambiente onde os recursos de hardware, o ambiente de programação e o conjunto de programas e protocolos sejam homogêneos. Um ambiente homogêneo torna o desenvolvimento de aplicações de grande escala mais fácil e acessível. A seguir são apresentados dois ambientes para a computação em nuvem – *VMWare VSphere*, que é proprietário e o *Eucalyptus*, que é *open-source* – ambos tentam oferecer a homogeneidade e as facilidades esperadas pelo consumidor.

2.2.3.1. Eucalyptus

O *framework Eucalyptus* oferece um IaaS para computação em nuvem, composto de vários componentes que interagem entre si através de interfaces bem definidas [Eucalyptus 2010]. A nuvem implementada pelo *Eucalyptus* aborda: agendamento e instanciação de máquinas virtuais (*Virtual Machine* - VM), armazenamento de dados e imagens de VMs, interfaces de administração e de consumidor para a nuvem, construção de redes virtuais, e definição e execução de SLAs.

O *Eucalyptus* implementa ambiente operacional para a nuvem que é independente do tipo de hipervisor – que atualmente pode ser Xen ou KVM. *Eucalyptus* foi projetado para ser o menos intrusivo possível, sendo bastante modular, baseado em padrões da indústria e com mecanismos de comunicação independentes de linguagem. A interface externa do *framework* é baseada em uma API desenvolvida pela *Amazon*. Adicionalmente,

a nuvem computacional fornece uma rede virtual sobreposta (*overlay*) que isola o tráfego de diferentes consumidores e permite que dois ou mais *clusters* (agrupamentos de máquinas físicas) pareçam pertencer à mesma rede local.

O *framework* utiliza emulação das interfaces SOAP (*Simple Object Access Protocol*) e *Query* do *Amazon EC2* (*Amazon Elastic Compute Cloud*), permitindo que os consumidores iniciem, controlem, acessem e finalizem VMs. Os consumidores interagem com o *Eucalyptus* utilizando as mesmas ferramentas e interfaces utilizadas para efetuar a interação com o *Amazon EC2*. Cada componente do sistema é implementado como um serviço *web* independente. Esta abordagem possui os seguintes benefícios: cada serviço *web* expõe um documento WSDL (*Web Services Description Language*) bem definido que permite a geração de uma API para qualquer linguagem; características já implantadas em serviços *web* podem ser utilizadas para prover comunicação segura entre os componentes.

Em uma nuvem *Eucalyptus* existem quatro componentes de alto nível (Figura 2.2), cada um com sua interface de serviço *web* [Nurmi et al. 2009]: 1) *Node Controller* – NC; 2) *Cluster Controller* – CC; 3) *Storage Controller* – *Walrus* e 4) *Cloud Controller* – CLC.

Node Controller

O *Node Controller* (NC) é executado em todo nó que hospeda uma VM, pesquisando e gerenciando o sistema hospedeiro e o hipervisor do nó, e respondendo às solicitações do *cluster controler* (evento *ger*, Figura 2.2). O NC consulta os recursos físicos do nó (e.g. número de núcleos, tamanho da memória, espaço disponível em disco, etc.) assim como informações sobre o estado das instâncias das VMs. As informações coletadas são propagadas para o *cluster controler* em resposta a solicitações de informações. Mediante a verificação da autorização e depois da confirmação da disponibilidade dos recursos, o NC executa as solicitações com o auxilio do hipervisor. Para iniciar uma instância de uma VM o NC executa uma cópia dos arquivos referente a imagem da mesma para o nó local – a partir de um repositório de imagens remoto ou de um *cache* local – criando um novo *endpoint* na rede virtual sobreposta e informando o hipervisor para inicializar tal instância. Para finalizar a instância, o NC informa o hipervisor para finalizar a VM, desfazer a rede e limpar os arquivos associados com tal instância.

Cluster Controller

O *Cluster Controller* (CC) geralmente é executado em uma máquina que é a porta de entrada (*front-end*) para o *cluster*, ou qualquer máquina que possua conectividade de rede com ambos os nós NC e CLC. Muitas das operações do CC são similares as operações do NC, porém, o volume de operações é maior. As funções primárias do CC são (evento *ad*, Figura 2.2): agendar solicitações para a execução de instâncias em NCs específicos, controlar a rede virtual sobreposta e recuperar/enviar informações sobre um conjunto de NCs. Quando o CC recebe uma solicitação para executar um conjunto de instâncias, este verifica cada NC (evento *ger*) e envia solicitações de execução de instâncias para o primeiro NC que tiver recursos livres suficientes para hospedar a instância. Quando o CC recebe uma solicitação para descrever os recursos (evento *ad*), este recebe uma lista

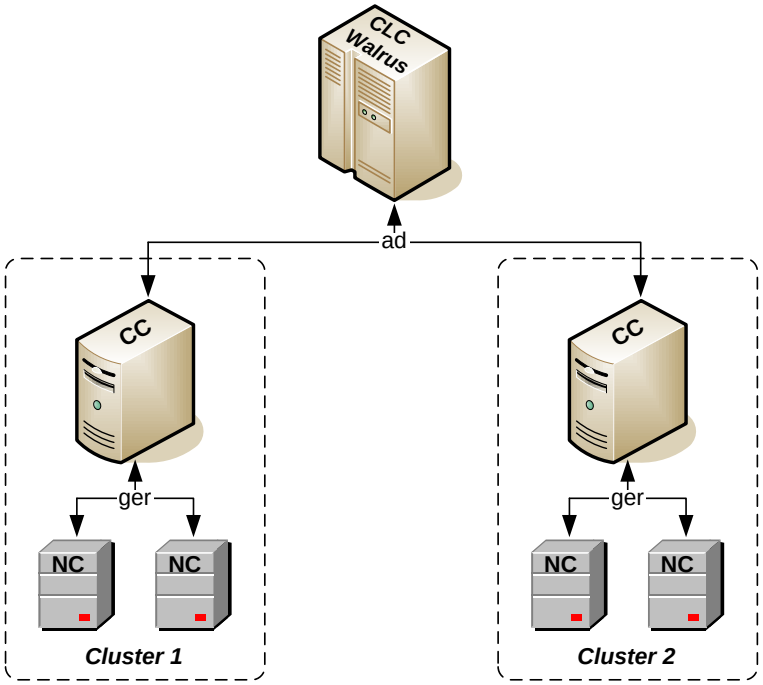

Figura 2.2. Visão geral do ambiente Eucalyptus.

das características do recurso desejado (e.g. núcleos, memória, disco) descrevendo os requisitos necessários para uma determinada instância. Com esta informação o CC calcula quantas instâncias simultâneas de um tipo específico podem ser executadas em sua coleção de NCs, reportando este número para o CLC.

Cloud Controller

Os recursos virtualizados que compõem a nuvem *Eucalyptus* são expostos e gerenciados pelo *Cloud Controller* (CLC). O CLC (evento *ad*, Figura 2.2) possui uma coleção de serviços web que são agrupados de acordo com seus papeis em três categorias: 1) *Resource Services*: executa a arbitragem do sistema para a alocação de recursos, permite que usuários manipulem propriedades de VMs e redes, e monitora os componentes do sistema e os recursos virtuais; 2) *Data Services*: administra dados do consumidor e do sistema *Eucalyptus*, fornece um ambiente consumidor personalizável para o estabelecimento das propriedades de alocação dos recursos; 3) *Interface Services*: fornece interfaces para os usuários, tratamento de autenticação e tradução de protocolos, e expõe as ferramentas de gerenciamento do sistema.

Storage Controller

O *Storage Controller* (*Walrus*) é um serviço de armazenamento de dados que estende as tecnologias padrão para serviços *web* (*Axis2*) tendo compatibilidade de interface com o *Amazon S3* (*Simple Storage Service*). *Walrus* implementa a interface REST (*Representatio-*

nal State Transfer, algumas vezes chamada de *Query*) assim como interfaces SOAP, ambas compatíveis com o S3. O armazenamento possui dois tipos de funcionalidades: 1) criar *streams* de dados fluindo para dentro/fora da nuvem, assim como a partir de instâncias inicializadas em nós; 2) funcionar como um serviço de armazenamento para imagens de VMs. Os arquivos que são utilizados para instanciar VMs em nós podem ser enviados para o *Walrus* e acessados a partir dos nós.

No projeto *Eucalyptus* a solução de rede utilizada para as instâncias de VMs aborda: conectividade, isolamento e desempenho. Toda VM controlada pelo *Eucalyptus* deve ter conectividade de rede com as outras e pelo menos uma das instâncias que fazem parte de um conjunto de VMs deve ter conectividade com a Internet. Com isto, o proprietário do conjunto pode efetuar *login* na instância conectada com a Internet e controlar as demais VMs. Os usuários possuem acesso de administrador nas respectivas VMs instanciadas e nas interfaces de rede. Esta funcionalidade pode causar preocupações à segurança, sendo que sem os devidos cuidados, a VM de um usuário pode adquirir qualquer endereço IP do sistema, causando interferência na rede do sistema ou em outras VMs que estiverem compartilhando o mesmo recurso físico. Assim, em uma nuvem compartilhada por diferentes usuários, VMs pertencentes a um único domínio devem ser capazes de se comunicar, porém VMs pertencentes a outros domínios devem ficar isoladas.

O *Eucalyptus Cluster Controller* (CC) é responsável por configurar e desfazer instâncias das interfaces de redes virtuais em três modos distintos: 1) neste modo o CC incorpora a interface de rede das VMs diretamente numa ponte *Ethernet* implementada em software – que está conectada a rede da máquina física. Isto permite ao administrador tratar pedidos DHCP da rede de VMs do mesmo modo que são tratados os pedidos DHCP que não fazem parte do *Eucalyptus*; 2) neste modo o administrador define endereços estáticos para o par IP/MAC – cada nova instância criada pelo sistema recebe um par IP/MAC livre, que é liberado quando a instância é finalizada. Nos modos 1 e 2, o desempenho de comunicações entre VMs é similar ao nativo – quando as VMs estão sendo executadas no mesmo *cluster* (*overhead* no desempenho pode ser imposto pelo hipervisor), porém neste caso não é provido o isolamento de rede entre VMs; 3) neste modo o *Eucalyptus* gerencia e controla as redes das VMs, fornecendo isolamento de tráfego entre VMs, definição das regras de entrada entre conjuntos lógicos de VMs e a atribuição dinâmica de endereços de IPs públicos para VMs durante o *boot* ou em tempo de execução.

2.2.3.2. VMware VSphere

O *VSphere* é vendido pela empresa *WMware* como um sistema operacional para computação em nuvem [VMWare Inc 2010]. *VSphere* é baseado no *VMware ESX/ESXi* que consiste em dois componentes interagindo entre si para fornecer um ambiente de virtualização robusto e dinâmico: o *Service Console* e o *VMkernel*.

O *Service Console* possui as funções de sistema operacional, usado para interagir com o *VMware ESX* e as máquinas virtuais que estão sendo executadas no servidor físico. O *Service Console* é derivado do Linux e inclui serviços encontrados em SOs tradicionais – *firewall*, agentes *Simple Network Management Protocol* (SNMP) e servidor *web*. Esse componente, apesar de definido pela *VMWare* como um SO de nuvem, apenas inclui os

serviços de SO para suportar a virtualização. O *Service Console* fornece acesso ao segundo componente, o *VMkernel* – base real do processo de virtualização. O *VMkernel* gerencia o acesso das máquinas virtuais ao hardware subjacente, escalonando o acesso ao processador e gerenciando a memória.

O *VMware ESX/ESXi* é um hipervisor do tipo nativo (*bare-metal*), instalado diretamente sobre o hardware e dividindo este com VMs que podem ser executadas simultaneamente, compartilhando os recursos físicos do servidor. Cada VM representa um sistema completo com processador, memória, rede, armazenamento, BIOS etc., podendo executar um SO tradicional com suas respectivas aplicações.

O *VMware vCenter Server* (*vCS* na Figura 2.3) é um utilitário para o gerenciamento centralizado de todos os *hosts ESX/ESXi* e suas respectivas VMs (evento *ger*). O aplicativo *vCenter Server* é baseado em *Windows* e permite aos administradores de TI: implantar, gerenciar, monitorar, automatizar e proteger a infraestrutura virtual. O banco de dados de suporte utilizado pelo *vCenter Server* – que pode ser o *Microsoft SQL Server* ou *Oracle* – armazena todos os dados sobre os *hosts* e as VMs.

O *VMware vSphere Client* (*vSC* na Figura 2.3) é uma aplicação baseada em *Windows* que permite o gerenciamento direto de *hosts ESX/ESXi* (evento *dir*) ou via *vCenter Server* (evento *ad*). O *vSphere Client* é uma interface gráfica usada para o gerenciamento de tarefas cotidianas e para a configuração da infraestrutura virtual. Se o *vSphere Client* for utilizado para se conectar diretamente a um *host ESX/ESXi*, esse exige o uso de uma conta de usuário daquele *host*, enquanto se for usado o *vSphere Client* para se conectar a um *vCenter Server* é exigida uma conta no servidor *Windows*.

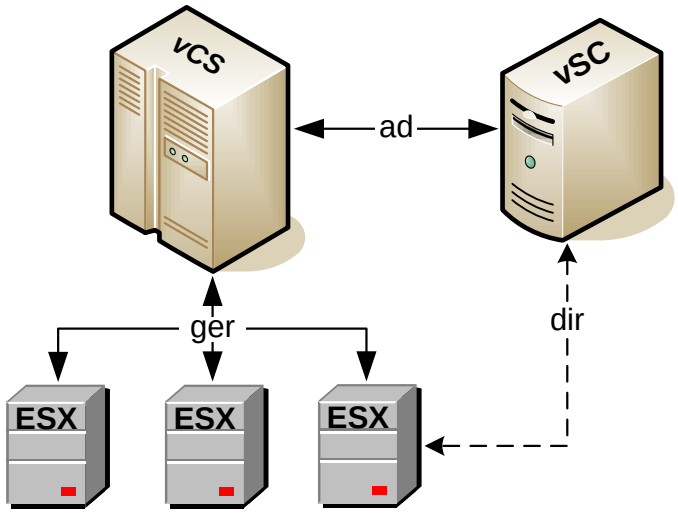

Figura 2.3. O ambiente VMware VSphere.

Todas as tarefas de gerenciamento disponíveis na conexão direta com um *host ESX/ESXi* estão disponíveis quando é feita a conexão com um *vCenter Server*, porém, o oposto não é verdadeiro. As funcionalidades de gerenciamento disponíveis através de um *vCenter Server* são mais significativas e superam as funcionalidades de se conectar diretamente a um *host ESX/ESXi*.

2.2.4. Principais diferenças entre computação em nuvem e em grid

A computação em nuvem não é um conceito completamente novo, estando relacionada com *grid*, *utility computing*, *cluster* e sistemas distribuídos [Foster et al. 2008, Zhang et al. 2010]. O modelo de *grid* está relacionado ao modelo de nuvem, porém, não necessariamente como seu antecessor [Nurmi et al. 2009]. Computação em nuvem e *grid* compartilham uma visão semelhante nos seguintes aspectos: a orientação a serviço, redução de custos computacionais, aumento da confiabilidade e da flexibilidade, e terceirização de tarefas de processamento. Ambas as abordagens podem ser úteis ao mesmo grupo de consumidores (e.g. pesquisadores executando computação paralela fracamente acoplada).

Apesar das semelhanças, *grid* e computação em nuvem diferem principalmente na arquitetura, pois em *grid* os pedidos de consumidores individuais podem (e deveriam) consumir grandes frações do *pool* total de recursos. Em computação em nuvem, frequentemente, se limita o tamanho de uma solicitação individual para ser uma pequena fração da capacidade total disponível no *pool* de recursos, porém tendo como objetivo ser escalável para atender um grande número de consumidores simultâneos. Nuvem e *grid* também compartilham problemas semelhantes, tendo dificuldades para: gerenciar grandes instalações, definir métodos para que consumidores possam procurar e interagir com novos recursos e serviços, e implementar computação paralela capaz de utilizar os recursos e serviços [Foster et al. 2008].

Os principais fatores que contribuem para a ampliação do interesse na computação em nuvem são: 1) aumento do poder computacional e capacidade de armazenamento; 2) crescimento exponencial na quantidade de dados gerados em pesquisas científicas e disponíveis na Internet, sendo constantemente publicados e recuperados; e 3) ampla adoção de *services computing* e aplicações Web 2.0.

Em ambientes de computação em nuvem temos a possibilidade de comprar acesso sob demanda para centenas de computadores em dezenas de *data centers* espalhados pelo mundo (e.g. *Amazon*, *Google*, *Microsoft*). Computação em nuvem é escalável, podendo ser vista como uma entidade que entrega diferentes níveis de serviço para seus consumidores. Essa é impulsionada pelo crescimento econômico, fornecendo serviços que podem ser dinamicamente configurados e entregues sob demanda.

A abordagem de *grid* surgiu com o intuito de resolver problemas computacionais de grande escala, usando uma rede de máquinas comuns que compartilham recursos. Ou seja, uma infraestrutura distribuída que fornece recursos de armazenamento e processamento. Para suportar a criação de organizações virtuais, por exemplo, *grids* fornecem *middlewares*, ferramentas e um conjunto de protocolos padrão que permitem a construção de serviços. Interoperabilidade e segurança são as principais preocupações em *grid*, visto que os recursos podem vir de diferentes domínios administrativos, possuírem políticas de uso globais/locais dedicadas, diversas configurações para o *hardware*, *software* e plataforma, com disponibilidade e capacidade variadas [Pinheiro Jr e Kon 2005].

A computação em nuvem implementa a abordagem proposta pelo modelo de *utility computing*: um modelo de negócio em que recursos computacionais – processamento, armazenamento – são empacotados e contabilizados como um serviço público que está

sendo consumido em nossas residências – e.g. energia elétrica ou água [Zhang et al. 2010]. No modelo de negócios utilizado pela computação em nuvem o consumidor paga ao provedor pelo consumo de determinado software e não pela licença do mesmo. O modelo confia no crescimento econômico para baixar os preços e aumentar os lucros.

A *Amazon* fornece o *Compute Cloud EC2* (cobrado com base no tempo de consumo) e o *Data Cloud S3* (cobrado por consumo de Gigabytes por mês). A transferência de dados é cobrada por consumo de Terabytes por mês. O modelo de negócios utilizado em *grid* é geralmente orientado a projetos, no qual usuários possuem unidades de serviço que podem ser consumidos – horas de processador, por exemplo. Quando uma instituição se associa ao *Teragrid* com seus recursos, por exemplo, ela tem conhecimento de que outros integrantes da comunidade podem utilizar os seus recursos.

No que diz respeito a federação de infraestruturas distintas, *grid* utiliza uma abordagem baseada em *middleware* como maneira de fornecer federações de recursos entre domínios administrativos cooperativos, porém separados geograficamente. Em nuvem, os serviços são distintos e não federados. Um provedor de computação em nuvem geralmente é operado por uma única entidade, com autoridade administrativa suficiente para estabelecer configuração uniforme e políticas adequadas.

A utilização de um *middleware* de *grid* juntamente com o modelo de computação em nuvem também pode ser feita, realizando assim tarefas de propósitos gerais em ambientes virtualizados [Caron et al. 2009]. Ou seja, a nuvem oferece recursos computacionais sob demanda enquanto o *middleware* de *grid* pode ser utilizado para gerenciar os recursos.

Muitos *grids* utilizam um modelo baseado em agendamento (*batch-scheduled*) em que um gerente de recursos locais (e.g. PBS, *Condor*) administra os recursos computacionais num determinado ambiente. Os usuários submetem trabalhos (e.g. *batch jobs*) solicitando algum recurso. Os trabalhos, geralmente, são agendados e enfileirados para processamento posterior, visto que alguns *grids* não suportam aplicações interativas. Para certos tipos de trabalho as decisões a serem tomadas a cerca do agendamento são muito caras em termos computacionais ou de tempo. Em computação em nuvem, os recursos computacionais são compartilhados por todos os usuários simultaneamente.

Um dos principais desafios para ganhar escalabilidade de maneira eficiente é a localização dos dados em relação aos recursos computacionais disponíveis. Para ter uma boa escalabilidade em computação em *grid* e nuvem, os dados devem ser distribuídos entre muitos computadores, sendo que o processamento deve ser executado visando reduzir custos de comunicação. Porém, *grid* geralmente utiliza sistemas de arquivos compartilhados (e.g. NFS, GPFS), que não consideram a questão de proximidade para fazer o armazenamento de dados.

A virtualização é um item indispensável para a computação em nuvem, fornecendo abstração e encapsulamento de dados e aplicações em um determinado domínio. Em geral, *grids* possuem um modelo de confiança diferente, no qual é utilizado delegação de identidade, para poder acessar recursos em diferentes domínios de um *grid* (e.g. *Ganglia*). Os recursos em *grid* não são abstraídos e virtualizados. Em nuvem, diferentes níveis de serviço são oferecidos, neste ambiente o usuário tem acesso a uma API, sendo que os recursos de baixo nível ficam escondidos (principalmente nos modelos SaaS e PaaS). As

informações retornadas para o usuário são limitadas, não fornecendo muitos detalhes sobre o estado do recurso [Foster et al. 2008].

Em computação em nuvem, trilhas de auditoria deixadas pelos processos podem ser utilizadas para rastrear a execução de um serviço desde sua fonte de dados, utilização de dados intermediários e procedimentos aplicados. Essa informação é vital para o entendimento, a descoberta, a validação de dados e processos. Estas informações também são úteis para encontrar erros na execução de fluxos de trabalho, validar ou invalidar resultados e servir de guia para projetos futuros. Em *grids*, o gerenciamento deste item, geralmente, está embutido em sistemas de fluxo de trabalho (e.g. *Chimera*, *Swift*) ou é provido como um serviço autônomo (e.g. *PreServ*). A aplicação deste tipo de gerenciamento em nuvem é de suma importância, visto que esta pode se expandir entre vários provedores de serviço, diferentes plataformas, políticas de acesso e camadas de hardware e software.

Grids têm por objetivo o processamento científico de grande escala, abrangendo e gerenciando uma grande quantidade de recursos (recursos que podem ser heterogêneos e instáveis). O modelo de programação MPI (*Message Passing Interface*) é o mais comumente utilizado – tarefas que usam a memória local da máquina durante o processamento e se comunicam com outras tarefas através do envio/recebimento de mensagens. Linguagens de coordenação permitem que componentes heterogêneos troquem dados, oferecendo facilidades para a estruturação dinâmica de componentes distribuídos (e.g. *Linda*). O sistema de fluxo de trabalho permite a composição dos passos individuais em um gráfico de dependência (e.g. *Swift*).

MapReduce é um modelo de programação paralela que fornece um sistema em tempo de execução para o processamento de grandes conjuntos de dados. Este é baseado nas funções "map" (aplica operações de mapeamento em um conjunto de itens) e "reduce" (executa a divisão de um conjunto de itens) [Grossman 2009]. O *MapReduce* particiona automaticamente dados de entrada e agenda a execução de programas em *clusters* de máquinas. Essa abordagem pode utilizar as máquinas virtuais fornecidas pela computação em nuvem.

Grids geralmente suportam aplicações do tipo HPC (*high performance computing*) e HTC (*high throughput computing*). Esta infraestrutura também suporta *gateways* científicos, que são *front-ends* para uma variedade de aplicações que podem ser fraca ou altamente acopladas. As aplicações para a nuvem caracterizam-se por serem fracamente acopladas, orientadas a transação e interativas (*grids* geralmente usam processamento em *batch*). As tecnologias Web 2.0 e os navegadores *web* têm um papel central na interação dos usuários com as nuvens computacionais.

A maioria das nuvens computacionais abrange *data centers* dedicados, que fazem parte da mesma organização, sendo que as configurações de hardware/software e as plataformas de suporte são homogêneas se comparadas aos ambientes de *grid*. Assim, a interoperabilidade pode se tornar um sério problema para comunicação entre *data centers*, interações que cruzam domínios administrativos etc.

Grids já são construídos considerando a heterogeneidade e o dinamismo dos recursos, onde cada domínio possui sua própria administração, operando de forma autônoma. Assim, aspectos de segurança já foram planejados desde a infraestrutura básica para supor-

tar: *single-sign-on*, delegação, privacidade, integridade, alocação coordenada de recursos, e reserva e compartilhamento de recursos. Ambientes de computação em nuvem ainda estão desenvolvendo estes aspectos.

Computação em *grid* fornece protocolos e serviços em cinco camadas diferentes [Foster et al. 2008]: 1) *fabric layer*: fornece acesso a diferentes tipos de recursos – processamento, rede, armazenamento. Geralmente conta com um gerente de recursos locais (e.g. PBS, Condor), componentes de propósito geral (e.g. GARA), e serviços especializados para o gerenciamento de recursos (e.g. Falkon); 2) *connectivity layer*: define os protocolos de comunicação e autenticação; 3) *resource layer*: define os protocolos para publicação, descoberta, negociação, monitoramento, contabilização e pagamento das operações compartilhadas pelos recursos (e.g. GRAM, gridFTP); 4) *collective layer*: captura interações entre coleções de recursos, serviços de diretório (e.g. MDS), agendamento e intermediação de serviços (e.g. Condor-G, Nimrod-G), sistemas de programação para grid (e.g. MPICH), e políticas para recursos (e.g. CAS); 5) *application layer*: engloba aplicações do usuário construídas em cima dos protocolos e APIs, operando em ambientes de organizações virtuais (e.g. *National Virtual Observatory*, *Teragrid Science gateway*).

Nuvens computacionais são normalmente vistas como um grande *pool* de recursos computacionais, podendo ser acessada via protocolos padrão através de uma interface abstrata – recursos e serviços podem ser fornecidos através de interfaces de Serviços Web. O ambiente de nuvem é projetado para resolver problemas computacionais na escala da Internet, sendo composta basicamente de quatro camadas [Foster et al. 2008]: 1) *fabric*: em nível de hardware – IaaS; 2) *unified resource*: recursos abstraídos ou encapsulados (e.g. virtualização) – IaaS; 3) *platform*: ferramentas especializadas, *middleware* e serviços – PaaS; 4) *application*: softwares executados na nuvem – SaaS.

2.2.5. Problemas em Aberto

Um dos obstáculos para a adoção massiva da computação em nuvem como um modelo de negócio é o capital que empresas tradicionais já investiram no passado na aquisição de hardware e licenças de software. Combinando estes fatores com os sistemas que foram desenvolvidos para serem executados nas infraestruturas já estabelecidas – sistemas legados – se chega a uma realidade onde parte ou a totalidade dos sistemas precisa ser re-projetado/reescrito para poder utilizar as APIs fornecidas pela computação em nuvem. Este problema existe, por exemplo, no dias atuais em grandes organizações do setor bancário, que tem sistemas do século XX desenvolvidos com tecnologias legadas (como ADABAS ou NATURAL, por exemplo) – dificilmente estes sistemas serão portados para outra plataforma. Adicionalmente, pode não haver uma reposição um-para-um para todos os sistemas sendo executados dentro dos limites organizacionais, portanto há casos em que as aplicações teriam que ser repensadas/desenvolvidas.

Andando na contra-mão da ampla adoção de computação em nuvem há o custo do hardware que diminuiu consideravelmente nos últimos anos, o que pode tornar interessante a utilização de computação em nuvem, mas em ambiente privado. Atualmente, empresas tradicionais têm gerado grandes quantidades de dados, hospedados em bancos de dados proprietários como o *MS-SQL Server* ou *Oracle* ou ainda estes dados estão armazenados em um sistema integrado de gestão empresarial (ERP) como o *SAP ERP*, por exemplo.

Empresas líderes de mercado em segmentos importantes ainda não possuem produtos suficientemente desenvolvidos para serem implantados na nuvem computacional ou não fornecem meios para portar os dados do ambiente tradicional para a nuvem computacional. Aparentemente, aspectos importantes da migração do ambiente tradicional para o ambiente de computação em nuvem ainda não estão bem definidos.

À medida que mais aplicações são transferidas para a nuvem computacional, mais largura de banda é necessária para transportar dados entre os provedores e os consumidores dos serviços fornecidos pela nuvem. Em vários lugares do mundo a infraestrutura de Internet disponível ainda é muito deficitária, porém quanto mais perto do provedor o consumidor estiver melhor é a qualidade de serviço que pode ser obtida. Assim, investimentos e estudos em novas tecnologias de rede são necessários, pois se não houver QoS na conexão de Internet para melhorar a taxa de transferência de dados e tempos de resposta aceitáveis para os consumidores, a computação em nuvem poderá ser prejudicada pela infraestrutura da Internet. Adicionalmente, muitas tarefas não podem ser executadas de maneira fácil devido as limitações do modelo pedido-resposta *stateless* (sem estado) do protocolo HTTP. Assim, para tratar todos os tipos de interações na nuvem – como é feito em aplicações de uma empresa tradicional – melhorias no protocolo de comunicação são necessárias.

Frameworks como o *MapReduce* [Grossman 2009] e suas implementações (e.g. *Hadoop*, *Dryad*) são projetados para o processamento distribuído de tarefas intensivas – altamente dependente dos dados. Estes *frameworks* geralmente operam em sistemas de arquivos sobre a Internet (e.g. GFS, HDFS). Estes sistemas de arquivos são diferentes dos sistemas de arquivos distribuídos tradicionais na sua estrutura de armazenamento, padrão de acesso e interface de programação. Assim, introduzem problemas de compatibilidade com sistemas de arquivos e aplicações legados. O desempenho e o consumo de recursos de uma tarefa *MapReduce* é altamente dependente do tipo da aplicação. Um dos desafios inclui a modelagem de desempenho de tarefas *Hadoop* (*online* ou *offline*) e o agendamento adaptativo em condições dinâmicas.

A implantação de métodos e ferramentas que sejam fáceis e eficientes para executar o procedimento de obtenção de imagens instantâneas, congelamento ou re-inicialização (*snapshot/restart*) de instâncias de VMs incentiva a conservação dos recursos computacionais. O ideal é que seja possível automatizar este processo para alcançar o máximo de economia, por exemplo, se uma VM está executando atividades não interativas e com pouca prioridade deve ser possível congelá-la automaticamente com base em parâmetros de configuração. Tais ferramentas, além de economia de energia e aluguel da infraestrutura de hardware devem proporcionar economias no uso de softwares, pois o paradigma de controle de licenças de softwares está mudando para um modelo de pagamento de acordo com o consumo. Algo como *pay-as-you-go* oferecido pela *Amazon* no EC2 para uso de *Windows Server*, *Windows SQL Server*, *IBM DB2 Express* e *IBM WebSphere*, por exemplo [Wang et al. 2010b].

A virtualização pode proporcionar benefícios significativos para a computação em nuvem, permitindo a migração de VMs para balanceamento de carga, por exemplo. Adicionalmente, migração de VMs permite um esquema de provisionamento robusto e sensível as características do ambiente [VMWare Inc 2010]. Porém, detectar carga de trabalho intensa e inicializar o processo de migração e reconfiguração das conexões

do consumidor com a VM necessita de mecanismos complexos. Ou seja, o sistema de gerenciamento deve ser capaz de responder rapidamente as mudanças bruscas de cargas de trabalho das VMs no IaaS.

A consolidação de servidores é uma abordagem efetiva para maximizar a utilização de recursos, enquanto minimizando o consumo de energia no ambiente de nuvem. A migração de VM em execução (*Live VM migration*) é utilizada frequentemente para consolidar (agrupar) vários servidores pouco utilizados em um único servidor com boa utilização. Com isto, os servidores que ficam sem utilização podem ser colocados em estado de espera, economizando energia [Etsion et al. 2009]. Porém, a consolidação de servidores de maneira próxima do ideal é frequentemente considerada um problema de otimização *NP-hard*.

Muitos ambientes de computação em nuvem comerciais são implementados em grandes *data centers*, administrados de modo centralizado (e.g. *Amazon*). Apesar deste projeto obter uma certa economia de escala e facilidade de gerenciamento, também introduz limitações como altos gastos com energia e um alto investimento inicial para construir o *data center*. A criação de pequenos *data centers* pode ser mais vantajosa se comparado aos grandes, pois o consumo de energia é menor, os sistemas de resfriamento são mais simples, a distribuição geográfica pode atender melhor os consumidores e o custo de construção dos mesmos é mais barato. As nuvens computacionais construídas com recursos voluntários ou uma mistura de recursos voluntários e dedicados – modelo híbrido – são mais baratas de se operar e mais adequadas para aplicações sem fins lucrativos, como a computação científica. Porém, esta arquitetura também traz desafios como o gerenciamento de recursos heterogêneos e a disponibilidade dos nós (*churn*).

2.3. Segurança em computação em nuvem

2.3.1. Fundamentos de segurança computacional

Todo o sistema computacional precisa ser protegido, porém é preciso analisar a sensibilidade dos dados que uma aplicação irá manipular para que a segurança seja dimensionada adequadamente [Landwehr 2001]. Ao longo dos anos, a segurança computacional passou por várias fases, inicialmente pretendia-se prevenir as violações de proteção, após esta fase o objetivo foi detectar e limitar as violações que não podiam ser prevenidas. Posteriormente, o foco foi tolerar os ataques, visando manter o fornecimento dos serviços. Estamos indo em direção a segurança comercializada como um serviço ou parte deste e neste caso temos que confiar em quem nos vende o serviço. Analogamente, a maneira como procedemos quando colocamos o nosso dinheiro em um banco – temos que confiar que o banco esteja integro e que os administradores não o levem a falência, porque se isto acontecer estaremos arruinados.

O esquema de segurança computacional necessita preservar as propriedades básicas: confidencialidade, integridade, disponibilidade, autenticidade e não repúdio. Além disto, alguns princípios devem ser considerados: responsabilização dos autores por suas ações, fornecimento do mínimo de privilégios possível para o desempenhar de uma atividade, minimização (da quantidade, do tamanho e da complexidade) dos componentes confiáveis do sistema e priorização do modo de operação seguro durante a implantação e utilização do sistema.

A segurança de sistemas computacionais envolve temas como políticas (conjunto de regras) de segurança e sua utilização em diferentes contextos – comercial, militar, doméstico etc. e a gerência de riscos. As políticas de segurança envolvem, por exemplo, definição de regras para: proteção do nível físico; contenção, recuperação de desastres, *backup*, preservação (durante o uso) e destruição (após o uso) de mídias; operação – envolvendo treinamento do usuário e registro de todas as ações de suporte; uso de criptografia e ciclos de vida de chaves; controle de acesso a sistemas e a recursos; não violação a leis e a ética, etc. A gerência de risco envolve a avaliação sistêmica e continuada dos níveis de segurança computacionais, avaliando os vários sistemas e aplicações de forma integrada para identificar vulnerabilidades, visando eliminá-las, mitigá-las ou tolerá-las [ISO 2005].

De maneira geral os mecanismos utilizados para dar suporte ao esquema citado acima são: definição de domínios de segurança vinculando os usuários a seus respectivos domínios, implantação de operações de autenticação, autorização, controle de acesso e auditoria, e a utilização de criptografia.

2.3.2. Gerenciamento de políticas de identificação e controle de acesso

Os serviços fornecidos pela nuvem computacional podem ser disponibilizados em qualquer local físico de abrangência da mesma, ou utilizando componentes de infraestrutura incompatíveis com o ambiente do consumidor. A gerência de um grande número de serviços (SaaS, PaaS, IaaS) e recursos físicos pode gerar um volume considerável de dados a ser administrada de maneira centralizada, pois será necessário coletar, armazenar, analisar e processar estes dados. Assim, a administração centralizada pode ser considerada impraticável, e portanto faz-se necessário instanciar serviços de gerenciamento distribuídos e fracamente acoplados (com baixa dependência funcional).

Para que as organizações consumidoras utilizem os serviços oferecidos pela nuvem é necessário a implantação de um modelo de gerenciamento seguro e confiável. O esquema a ser utilizado deve facilitar a inserção e remoção de usuários dos serviços oferecidos pela nuvem. A implantação de mecanismos de autenticação robustos e esquemas de delegação de direitos funcionando de maneira confiável são fundamentais para o correto gerenciamento de identidades e para a prestação de serviços em nuvens computacionais.

Serviços de identidade utilizados pela nuvem devem suportar a delegação de direitos administrativos, com isso o gerenciamento pode ser repassado aos administradores individuais de cada ambiente – SaaS, PaaS, IaaS – então cada administrador pode gerenciar contas dentro de seu próprio domínio.

Para fornecer acesso aos diferentes níveis de serviço, a organização consumidora pode utilizar um serviço de SSO (*Single Sign-On*) que faça parte de uma federação para autenticar os usuários das aplicações disponíveis na nuvem (Figura 2.4; evento *aut*). O provedor de SSO pode ser terceirizado, instanciado externamente a organização consumidora (Domínio B). O *OpenID* é uma opção quando a organização consumidora deseja ter o processo de identificação terceirizado [OpenID 2010]. No ambiente de computação em nuvem, a federação de identidades tem um papel fundamental para permitir que organizações consumidoras associadas se autentiquem a partir de um único ou simples *sign-on* (evento *ac*). Então, poderá acontecer a troca de atributos de identidades entre o provedor de serviço e o de identidade (evento *atr*). Padrões como a *WS-Federation* podem auxiliar

na federação de identidades para diferentes domínios administrativos [OASIS 2009b].

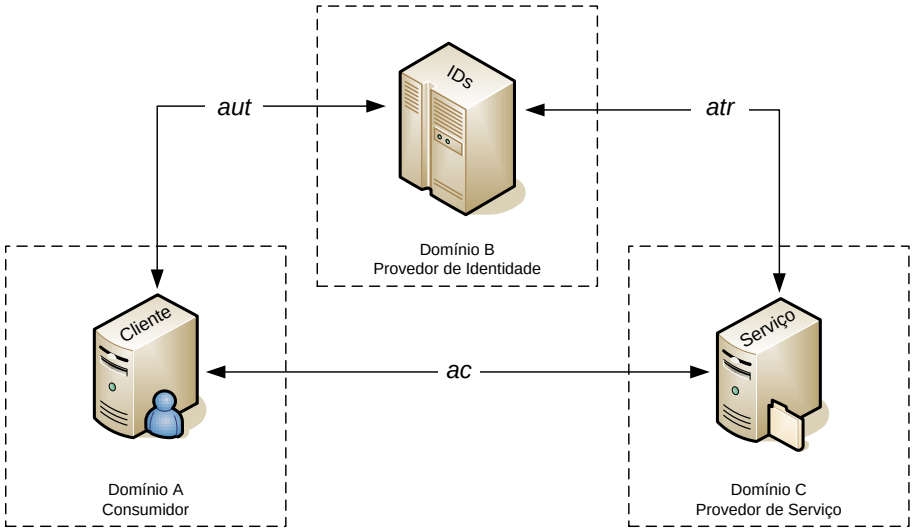

Figura 2.4. Federação de Identidades.

O nível de IaaS atende tipicamente administradores de tecnologia da informação. Assim, a divisão de atribuições dos usuários segue os requisitos de gerenciamento de privilégio de um *data center* comum – e.g. administrador do sistema, engenheiro de rede etc. O nível IaaS basicamente é voltado ao gerenciamento do ciclo de vida de servidores virtuais – interação com máquinas virtuais, envolvendo sua criação, destruição, inicialização, parada, exportação e importação.

No nível de IaaS, o provedor de computação em nuvem não tem controle sobre os serviços instanciados nas máquinas virtuais que autenticam seus usuários. Ou seja, é a organização consumidora (contratante) que decide como o serviço vai executar a autenticação dos seus usuários. Assim, o serviço pode aceitar credenciais de autenticação em vários formatos (e.g. SAML – *Security Assertion Markup Language* [OASIS 2005a], certificados X.509 [ITU-T 2000]). Certificados digitais são mais apropriados para transportar informações que não são alteradas frequentemente. Adicionalmente, esta abordagem necessita de uma infraestrutura de gestão de certificados (por exemplo, o serviço XKMS – *XML Key Management Specification* [W3C 2001]) e procedimentos para manter a integridade da informação.

Provedores de serviço nos níveis SaaS e PaaS geralmente oferecem serviços de autenticação acoplados as suas aplicações e plataformas. Alternativamente os provedores podem delegar a autenticação dos usuários para a organização contratante dos serviços. Com esta abordagem, o contratante pode autenticar seus usuários localmente, utilizando um serviço de identificação interno a organização e estabelecendo confiança com o fornecedor de serviço através da federação de identidades, por exemplo. Caso o usuário esteja agindo em seu próprio nome, o processo de autenticação pode ser centrado no usuário, utilizando algum tipo de identidade válida para a nuvem (e.g. *LiveID*).

Para o ambiente de nuvem o gerenciamento da autenticação deve fornecer suporte aos processos de criação e emissão das credenciais (e.g. senhas, certificados digitais,

credenciais dinâmicas) utilizadas pelos usuários da organização. Procedimentos de autenticação robustos (e.g. baseada em múltiplos fatores) podem não ser compatíveis com determinados serviços fornecidos pela nuvem. Consequentemente, a utilização de uma grande variedade de métodos de autenticação gerará carga administrativa adicional. O usuário dos serviços também precisa ter a usabilidade considerada, pois esse pode necessitar utilizar um conjunto de métodos para as aplicações internas a organização, e outro conjunto para acessar os serviços na nuvem. O mesmo desafio aplica-se aos provedores de computação em nuvem, pois o custo para suportar vários mecanismos de autenticação, acomodando as necessidades de consumidores usando mecanismos heterogêneos pode se tornar pouco atrativo para a entidade que mantém a nuvem. Neste caso, o ideal é a padronização dos mecanismos de autenticação para resolver estas limitações impostas pelas características de computação em nuvem.

O ambiente de computação em nuvem está sendo utilizado para hospedar vários tipos de serviços, e todos exigem garantias de segurança dos dados sendo processados e armazenados. Para tirar o máximo proveito de todo o poder oferecido pela nuvem computacional, as diferentes entidades – provedores e consumidores de serviços – que interagem com o ambiente necessitam de abordagens de segurança abrangentes e confiáveis. O particionamento do ambiente de computação em nuvem em diferentes domínios cria escopos de proteção reduzidos, regrando e limitando as interações entre as partes, classificando os tipos de serviços e recursos, facilitando as operações de gerenciamento, efetuando o balanceamento e a distribuição de carga etc. [Goyal e Mikkilineni 2009].

A definição de um sistema de segurança baseado em políticas é uma necessidade administrativa e de uso da nuvem computacional, pois é possível controlar o acesso e uso individual de cada usuário do ambiente [Yildiz et al. 2009]. Cada organização consumidora de serviços fornecidos pela nuvem precisa definir políticas para seus usuários. Os seguintes tipos de usuários devem ser considerados nas políticas: administrador, desenvolvedor e usuários finais da organização. Adicionalmente, práticas, processos e procedimentos de gerenciamento de identidade e acesso devem englobar os serviços oferecidos pela nuvem. O gerenciamento deve ser, preferencialmente, escalável, efetivo e eficiente para ambos, provedores e consumidores dos serviços [CSA 2010a].

Os perfis e as políticas de controle de acesso podem variar de acordo com o tipo de consumidor – organização ou usuário – de serviços da nuvem. Quando o consumidor é uma organização as políticas envolvem a definição de regras para a totalidade do domínio da organização consumidora, enquanto para usuários da organização as políticas precisam ser individualizadas. O perfil do usuário pode ser descrito como um conjunto de atributos utilizados pela nuvem para customizar o serviço e possivelmente restringir o acesso a outros serviços. O procedimento de controle de acesso utiliza informações sobre o perfil do consumidor para tomar decisões na escolha das políticas. Quando o consumidor é um usuário seus atributos são a única fonte de informações para o monitor de referência, sendo que as políticas foram definidas pela organização consumidora de computação em nuvem. Quando o consumidor representa uma organização, os atributos do perfil são obtidos de SLAs ou contratos – podendo envolver requisitos de QoS (*Quality of Service*).

A utilização de serviços dentro da nuvem cria a possibilidade das políticas de controle de acesso serem definidas em um lugar – por exemplo, internamente a organi-

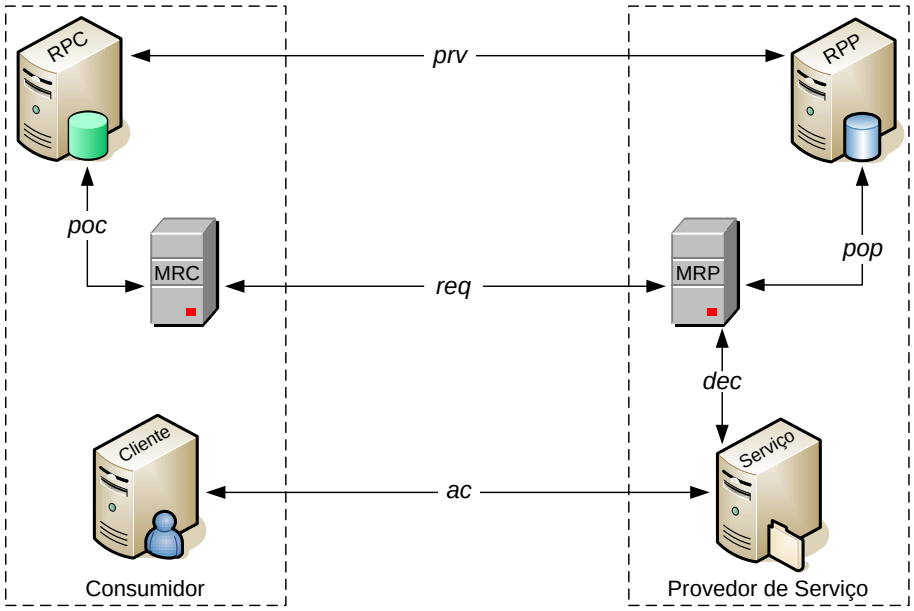

Figura 2.5. Provisionamento de políticas no provedor.

zação (Figura 2.5; evento *poc*; Repositório de Política do Consumidor – RPC) – e serem executas em outro (evento *pop*; Repositório de Política do Provedor – RPP). Isto é, as políticas definidas pela organização consumidora podem ser transferidas do repositório de políticas para o provedor de computação em nuvem que controla o uso dos serviços. Padrões de serviços web como a XACML – *eXtensible Access Control Markup Language* [OASIS 2005b] e a *WS-Policy* ([W3C 2010] são úteis nestes casos. Porém, mesmo com a utilização de especificações padronizadas, o consumidor e o provedor de computação em nuvem precisam utilizar a mesma semântica para que a interação entre as entidades dos diferentes domínios possa ocorrer de maneira transparente e segura – e.g. solicitações de acesso aos serviços (evento *ac*).

A transferência ou configuração de políticas pode ocorrer periodicamente, no modo *batch* ou sob demanda (*just-in-time*), quando será enviada concomitantemente com a solicitação de configuração vinda do Monitor de Referência do Provedor (MRP; evento *prv*). Para o modo *batch* a especificação SPML (*Service Provisioning Markup Language*) pode ser utilizada. Porém, se o modo *single sign-on* estiver ativo e o provedor de serviço de computação em nuvem for capaz de receber informações de políticas em assertivas SAML, sob demanda, então as informações podem ser transmitidas utilizando o *SAML profile of XACML* [OASIS 2005c]. Adicionalmente, se as decisões de autorização forem avaliadas externamente aos serviços hospedados na nuvem (i.e. no Monitor de Referência do Consumidor – MRC), o padrão XACML pode ser utilizado para expressar solicitações e respostas de avaliações de políticas (evento *req*).

O provisionamento (pré-configuração) de políticas para consumidores pode ser efetuado com a utilização da especificação SPML [OASIS 2006]. As políticas podem conter papeis de acesso pré-definidos: administrador, desenvolvedor e usuário final. O ambiente no nível PaaS tipicamente atende desenvolvedores de serviços. Os desenvolvedores, por exemplo, podem necessitar contas para teste de suas aplicações – necessitando de vários

usuários temporários para depurar os serviços. Observe que neste caso os usuários de teste teriam o perfil de usuário final, mas estariam sob o controle de desenvolvedores.

Os consumidores precisam ter certeza que os provedores de nuvem suportam suas necessidades e fornecem mecanismos de controle de acordo com a dinâmica exigida pelo ambiente. Basicamente, o provedor de computação em nuvem precisa: controlar o acesso de usuários aos serviços fornecidos pela nuvem – de acordo com as políticas definidas pelo consumidor; honrar os SLAs ou contratos de QoS estabelecidos com organização consumidora; controlar o acesso aos dados dos usuários; controlar acesso a área do sistema no nível de usuário e administrativo (privilegiado); manter as informações do perfil do usuário e as políticas de controle de acesso atualizadas; permitir a coleta de informações do perfil do usuário e das políticas de controle de acesso implantadas no provedor (para um determinado consumidor); fornecer meios de notificação para alterações em contas de usuários (e.g. criação, remoção, concessões de acesso), visando coibir a configuração de contas falsas ou modificação de direitos de acesso no provedor sem que o consumidor saiba; e fornecer trilhas de auditoria para o ambiente de cada consumidor – identificando atividades de gerenciamento e acesso, assim como a utilização de qualquer recurso para qual foram estabelecidas cotas de uso.

2.3.3. Riscos e ameaças em ambientes de nuvem computacional

As organizações devem avaliar o risco e as opções de segurança antes de mover seus sistemas e aplicativos para o ambiente de computação em nuvem. É necessário avaliar quais dados e serviços podem ser transferidos para o ambiente externo a organização se a nuvem for pública. Os principais tipos de ativos (*assets*) suportados pela nuvem são: dados, aplicações, processos e serviços. Estes ativos/recursos devem ser analisados para que se possa determinar sua importância para o negócio da organização. No processo de análise busca-se avaliar os impactos gerados caso algum requisito de segurança (confidencialidade, integridade ou disponibilidade) seja comprometido. As organizações podem mover integral ou parcialmente seus processos ou dados para o ambiente de computação em nuvem. Parte das transações e informações podem ser mantidas dentro do perímetro da organização – em ambiente privado [CSA 2009].

Entender os relacionamentos e dependências entre as diferentes camadas da computação em nuvem é fundamental para entender os riscos de segurança dos três principais modelos (Saas, PaaS e IaaS) de implantação de nuvem computacional. SaaS provê o conjunto mais integrado de funcionalidades de computação em nuvem, construídas de acordo com as necessidades dos usuários – possuindo o mínimo de extensibilidade (porque o serviço já está definido) para o consumidor e com um nível relativamente alto de segurança integrado ao serviço. PaaS destina-se a permitir que os desenvolvedores construam suas próprias aplicações em cima da plataforma – é mais flexível que o modelo SaaS – oferece ao consumidor funcionalidades pré-configuradas para que este possa escolher e usar. Como o conjunto de funcionalidades não é completo ou plenamente integrado, existe uma maior flexibilidade para se inserir camadas de segurança adicionais. IaaS provê poucas funcionalidades específicas para as aplicações, porém é flexível e extensível. Este provê poucas funcionalidades de segurança integradas – além da proteção da própria infraestrutura da nuvem. IaaS necessita que o sistema operacional, aplicações e conteúdo seja gerenciado e protegido pelo consumidor da nuvem computacional.

Os controles de segurança na computação em nuvem são, em sua maioria, iguais aos controles de qualquer ambiente de tecnologia da informação. Porém, de acordo com os modelos de serviço (SaaS, PaaS, IaaS), modo de operação – administração do ambiente – e tecnologias utilizadas para prover os serviços na nuvem, estes pode apresentar diferentes riscos para a organização quando comparado com as abordagens tradicionais. Na computação em nuvem se abre mão de alguns controles (físico, por exemplo) enquanto se mantêm as responsabilidades sobre o gerenciamento operacional.

Os domínios de segurança de um ambiente de computação em nuvem podem ser administrativo ou operacional. O domínio administrativo aborda assuntos relacionados a política e estratégia de negócio. O domínio operacional preocupa-se com a segurança dentro da arquitetura [CSA 2009]:

- **Domínio Administrativo**: aborda aspectos como: ações legais por violação de contrato, gestão de proteção de dados sensíveis quando o usuário ou o provedor podem ser os responsáveis por erros ou falhas e a influência que a localização física da nuvem pode trazer para os aspectos citados anteriormente – devido a diversidade de tratamento do assunto de acordo com as leis de diferentes países. O gerenciamento administrativo de risco pode ser descrito como a capacidade da organização gerenciar e medir o risco introduzido pela adoção do modelo de computação em nuvem [Kaliski Jr e Pauley 2010].

  Assuntos legais relacionados a utilização da nuvem envolvem a proteção da informação e sistemas computacionais e contramedidas para tratar as violações de segurança e privacidade – por intrusão, vazamento, revelação ou divulgação de dados protegidos. Os aspectos relativos a privacidade serão abordados na Seção 2.4.

  Gerenciar o ciclo de vida dos dados colocados na nuvem computacional está relacionado a identificação e controle de dados. Estes envolvem a definição de controles compensatórios que possam ser utilizados para tratar a perda do controle físico sobre os dados quando estes forem transferidos para a nuvem, e a definição de quem é a entidade responsável pela confidencialidade, integridade e disponibilidade dos dados.

  A utilização de serviços terceirizados precisa considerar a capacidade de mover dados e serviços de um provedor de computação em nuvem para outro, ou trazer os dados novamente para dentro da organização quando necessário. A interoperabilidade entre provedores e consumidores deve ser cuidadosamente avaliada para que a migração de dados não se torne um problema.

- **Domínio Operacional**: a computação em nuvem afeta o domínio operacional – os procedimentos utilizados para implementar a segurança, políticas de continuidade do negócio e recuperação de desastres [ISO 2005]. Assim, é necessário avaliar os possíveis riscos com a intenção de implementação de modelos de gerenciamento de risco adequados ao perfil de consumidores de serviços da nuvem computacional.

  A correta operação da tecnologia de virtualização na computação em nuvem é um item fundamental para mitigação do risco operacional. A utilização de máquinas virtuais possui os seguintes desafios associados [Ristenpart et al. 2009]: garantir que

o arrendamento compartilhado da máquina física não traga problemas de segurança para o consumidor (multilocação segura); garantir o isolamento (não interferência) entre VMs alocadas sobre uma mesma plataforma física (*co-residence*) e garantir monitoramento de vulnerabilidades em nível de hipervisor.

É indispensável definir métodos para detectar incidentes e fornecer notificações ou respostas com as respectivas contramedidas – abordando mecanismos que deveriam estar implantados tanto no nível do provedor quanto no nível de consumidor – possibilitando o correto tratamento e avaliação de incidentes, por exemplo, pode-se utilizar métodos de forense digital nestes casos.

2.3.3.1. Principais Ameaças

Com o modelo de computação em nuvem a informática passa a ser um utilitário que pode ser contratado e utilizado de acordo com a necessidade, sem que o consumidor (contratante) tenha que se preocupar com o gerenciamento da infraestrutura. Apesar dos benefícios que a computação em nuvem pode trazer, os consumidores estão preocupados com os riscos que a utilização de um ambiente novo pode representar para os ativos (*assets*) da organização. Em outras palavras, uma organização estará terceirizando um sistema pelo qual é responsável, por exemplo, sem ter total controle sobre o mesmo. Assim, a decisão de adotar a computação em nuvem precisa ser bem estudada para que o risco possa ser calculado e a computação em nuvem traga benefícios efetivos.

A segurança e a privacidade (Seção 2.4) são os principais desafios que podem impedir a ampla adoção da abordagem de computação em nuvem. Pois, falhas de segurança em qualquer um dos componentes podem impactar os demais componentes de segurança e consequentemente a segurança de todo o sistema poderá entrar em colapso [CSA 2010b].

Provedores de IaaS, por exemplo, oferecem para seus consumidores a ilusão de uma capacidade computacional ilimitada de processamento, largura de banda e armazenamento. O processo de registro para compra desse serviço é simples e pode ser feito por qualquer portador de um cartão de crédito válido. Como os registros podem ser anonimizados (tornados anônimos através do uso de logins - que identificam um usuário do sistema e não entidades no mundo real) e o uso do serviço é imediato após a contratação, malfeitores como *spammers*, programadores de *botnets* etc. são capazes de realizar atividades maliciosas com relativa impunidade [Provos et al. 2009, Zhao et al. 2009]. As soluções possíveis para mitigar os casos citados devem envolver processos de registro e validação de usuários mais rigorosos, aprimoramento da monitoração de fraudes no uso de cartão de crédito e monitoração do tráfego de rede do consumidor sem violar sua privacidade – monitoramento do trafego de informações de origem e destino públicos, por exemplo.

O sequestro de contas ou serviços não é um problema de segurança desconhecido, pois acontece em *phishing*, fraudes, exploração de vulnerabilidades de sistemas e aplicações etc., sendo que é uma prática comum usuários reutilizarem credenciais e senhas, amplificando o impacto deste tipo de ataque. Uma vez obtidas as credenciais de seu alvo, um indivíduo mal intencionado pode acompanhar as atividades e transações efetuadas pela conta de acesso, gerando uma diversidade de problemas (e.g. leitura, alteração e inserção de dados; redirecionamento de clientes para domínios falsos; subversão de instâncias de

serviços legítimos etc.). O compartilhamento ou a delegação de credenciais entre entidades deve ser evitado ou muito bem monitorado. O sistema deve implantar técnicas robustas de autenticação, preferencialmente baseadas em vários fatores (combinação de técnicas que exploram, por exemplo: algo que se é – uma característica física como a impressão digital; algo que se tem – um *token*; ou algo que se sabe – uma senha) e monitoramento para detectar atividades não autorizadas ou suspeitas – através da implantação de um sistema de detecção de intrusão, por exemplo.

Diferente das abordagens tradicionais de tecnologia da informação, a computação em nuvem oferece grande flexibilidade para os usuários visto que estes não precisam se preocupar com a complexidade de gerenciamento inerente a cada sistema (e.g. os banco de dados podem ser transferidos para *data centers* de grandes empresas especializadas). Porém, o gerenciamento dos dados em ambientes terceirizados nem sempre é confiável. Os usuários acabam ficando a mercê da disponibilidade e integridade provida pelos provedores de serviço de armazenamento (e.g. *Amazon Simple Storage Service* – S3). Assim, é necessária a utilização de modelos de armazenamento de dados seguros visando garantir a integridade dos dados dos consumidores [Wang et al. 2009].

Os dados de usuários e consumidores podem ser comprometidos de várias maneiras, por exemplo, informações que não possuem cópia de segurança podem ser eliminadas ou alteradas, os registros de um contexto podem ser desvinculados, o armazenamento pode ser feito em mídias não confiáveis, a chave de codificação pode ser perdida etc. O risco de comprometimento de dados aumenta na nuvem devido ao grande número de desafios inerentes as características arquiteturais e operacionais desse ambiente – implementação de controles de autenticação, autorização, auditoria e cifragem; falhas operacionais; problemas de jurisdição; confiabilidade do *data center* etc. Ambientes terceirizados ou externos necessitam de práticas e mecanismos de segurança rigorosos para garantir a segurança dos dados em trânsito, proteção dos dados utilizados em processos, gerenciamento do ciclo de vida de chaves, estabelecimento de regras contratuais com os provedores exigindo a correta destruição de dados que estão armazenados em mídias antes desta ser liberada para uso, estratégias para efetuar a cópia de segurança etc.

A arquitetura orientada a serviço (*Service Oriented Architecture* – SOA) e o modelo de computação em nuvem podem ser considerados serviços complementares [Zhang e Zhou 2009]. SOA compreende um conjunto de princípios e metodologias projetadas para facilitar a integração de sistemas e a comunicação independentemente da linguagem de desenvolvimento ou plataforma. Enquanto que a computação em nuvem foi projetada para permitir uso instantâneo e massivo da capacidade de processamento e armazenamento, por exemplo, sem que seja necessário investir em infraestrutura, treinamento de equipes, licenciamento de softwares etc.

Independentemente das diferenças nos propósitos de projeto de cada uma, a abordagem de computação em nuvem pode se relacionar com a arquitetura SOA na utilização de componentes como um serviço (*Components as a Service*, CaaS – SOA implementada através de padrões para serviços web). Assim, a computação em nuvem e SOA podem ser desenvolvidas independentemente, ou concomitantemente como atividades complementares visando provimento de um negócio que atenda a uma vasta gama de consumidores [Dawoud et al. 2010]. Para acessar recursos que disponibilizam interfaces

de serviços web, o protocolo SOAP (*Simple Object Access Protocol*) é o mais utilizado, junto com a *WS-Security* – uma das extensões padrão para proteger as mensagens em trânsito. Os ataques que visam afetar a segurança da XML (*Extensible Markup Language*) [Gruschka e Iacono 2009] podem ser mitigados com a utilização de especificações fornecidas por organizações como OASIS e W3C [Jensen et al. 2009].

A grande maioria dos provedores de computação em nuvem expõe um conjunto de interfaces para gerenciar e interagir com os serviços oferecidos (e.g. provisionamento, gerenciamento, orquestração, monitoramento etc.). A segurança e a disponibilidade de alguns serviços da nuvem dependem da segurança destas APIs, por exemplo, autenticação, controle de acesso, cifragem e monitoramento. As interfaces devem ser projetadas para se proteger de tentativas acidentais ou maliciosas de violação de políticas. O modelo de segurança das interfaces disponibilizadas pelo provedor deve ser cuidadosamente analisado e avaliado, assegurando-se que mecanismos consistentes de autenticação e controle de acesso estejam implementados.

O nível de acesso concedido aos funcionários do provedor de computação em nuvem poderia permitir que um indivíduo mal intencionado obtivesse dados confidenciais ou ganhasse controle sobre os serviços disponibilizados no provedor. A utilização de um único domínio de gerenciamento combinado com a falta de transparência dos processos e procedimentos aplicados pelo provedor da nuvem (e.g. ausência de políticas que envolvem os funcionários no procedimento de concessão de direitos e monitoramento de acessos a ativos físicos e virtuais) podem fazer deste ambiente um lugar hostil para processos e informações. Procedimentos de gestão rigorosos devem ser aplicados a toda a cadeia de provimento de serviços (*supply chain*) avaliando cuidadosamente todas as entidades que interagem direta ou indiretamente com o serviço (e.g. transparência na definição de políticas segurança da informação, boas práticas de gerenciamento, relatórios de conformidade, notificação de falhas etc.).

O nível IaaS serve de base para os demais modelos de fornecimento de serviço (PaaS e SaaS), sendo que a falta de segurança neste nível certamente afetará os modelos construídos sobre a mesma. O uso da virtualização permite que os provedores de computação em nuvem maximizem a utilização do hardware, comutando várias VMs (*Virtual Machine*) de consumidores em uma mesma infraestrutura física. Esta abordagem pode introduzir vulnerabilidades, pois geralmente executa-se a multilocação (*multi-tenancy*) – comutação das máquinas virtuais de consumidores distintos sobre o mesmo hardware. Assim, a VM de um consumidor poderia ser alocada no mesmo servidor físico de seu adversário ou concorrente. Esta abordagem traz um novo tipo de ameaça, pois o adversário poderia violar o isolamento entre as VMs – explorando vulnerabilidades que permitam se infiltrar no hipervisor, ou usar canais secundários (*side-channels*) visando obter acesso não autorizado a dados ou processos [Ristenpart et al. 2009, Dawoud et al. 2010].

Os fornecedores de IaaS prestam serviços compartilhando uma mesma infraestrutura física. Porém, componentes físicos do hardware como a memória *cache on die* da CPU não foram projetados para oferecer propriedades de isolamento para uma arquitetura com vários usuários (*multi-tenant* – multi-inquilinos). Para contornar essa limitação, o hipervisor faz a mediação do acesso entre o sistema operacional convidado e os recursos computacionais físicos. Porém, hipervisores podem apresentar falhas que permitam ao

sistema convidado obter níveis impróprios de controle e ou influência na plataforma subjacente. O isolamento (confinamento) das VMs deve ser assegurado para que um consumidor não viole o *address space* de outros consumidores sendo executados no mesmo provedor de computação em nuvem. Todo o processo de implementação, instalação e configuração do ambiente de nuvem deve seguir boas práticas de segurança, sendo que após esta fase o ambiente deve ser monitorado visando detectar mudanças ou atividades não autorizadas.

Um dos princípios da computação em nuvem é a redução dos custos com ativos/recursos de hardware/software e a respectiva manutenção associada aos mesmos, permitindo que as empresas concentrem-se em seus negócios. Esta abordagem tem benefícios financeiros e operacionais, mas deve ser cuidadosamente avaliada devido as preocupações com a segurança – principalmente, atualizações e a versão de um sistema, boas práticas e políticas de segurança. Modelos de segurança para a camada de IaaS podem ser utilizados como um guia para avaliar e reforçar a segurança do ambiente [Dawoud et al. 2010].

2.4. Privacidade e computação em nuvem

A segurança da informação se refere à proteção sobre as informações de uma determinada empresa ou indivíduo, isto é, aplica-se tanto às informações corporativas quanto às pessoais. Todavia, existe uma relação inversa quando se trata de privacidade e segurança, pois quanto maior a segurança coletiva, geralmente menor é a privacidade individual [Fischer-Hübner 2001]. Um exemplo dessa relação são os sistemas de monitoramento com vídeo em prédios e ambientes públicos.

Para [Sweeney 2002], a segurança computacional não implica em proteção à privacidade, pois embora mecanismos de controle de acesso e autenticação possam proteger as informações contra a divulgação, eles não tratam da propagação indireta da informação, nem de divulgações com base em inferências e correlações sobre informações extraídas de outras fontes. Esta seção apresenta os principais conceitos relacionados à privacidade e alguns mecanismos usados (ou propostos) para protegê-la em ambientes computacionais, com ênfase em computação em nuvem.

2.4.1. O conceito de privacidade

De acordo com [Shirey 2000], a privacidade pode ser definida como o direito de uma determinada entidade (normalmente um indivíduo), agindo em seu próprio nome, de determinar o grau de interação de suas informações com contexto onde se encontra inserida, incluindo o grau de comprometimento/disposição em divulgar essas informações para outras entidades. Existem basicamente três elementos na privacidade: o *sigilo*, o *anonimato* e o *isolamento* (ou solidão, o direito de ficar sozinho) [Fischer-Hübner 2001, Wright 2004]. O sigilo é um problema fortemente ligado à confidencialidade, o anonimato está relacionado à proteção da identidade do sujeito e o isolamento é o direito de ficar indisponível para outros indivíduos.

O direito à privacidade é um conceito consolidado em algumas áreas, como a médica, a jurídica e a fiscal. No contexto médico [Beaver e Harold 2004], a privacidade consiste na limitação do acesso às informações de um indivíduo, ao acesso ao próprio indivíduo ou à sua intimidade. Em outras palavras, é o direito do indivíduo não ter sua vida ou seus dados observados sem autorização. Para os juristas, privacidade é o direito

de ficar sozinho [Staples 2007]. A privacidade está atrelada à questão do anonimato, ou seja, à condição de um indivíduo ter suas informações pessoais protegidas [Shirey 2000]. A privacidade também pode ser vista como a capacidade de um usuário realizar ações em um sistema sem ser identificado.

Em geral, a noção de privacidade abrange três diferentes escopos [Fischer-Hübner 2001]: *Privacidade territorial*: proteção da região próxima a um indivíduo, como seu ambiente doméstico, local de trabalho ou espaços públicos; *Privacidade do indivíduo*: proteção contra interferências indesejadas, como buscas físicas (revistas), testes com drogas ou informações que possam violar aspectos morais do indivíduo; e *Privacidade da informação*: designa quando e como dados pessoais de um indivíduo podem ser coletados, armazenados, processados e propagados a terceiros.

A definição mais comum e aceita no mundo da informática diz que *a privacidade consiste nos direitos e obrigações dos indivíduos e organizações com relação à coleta, uso, conservação e divulgação de informações pessoais* [Mather et al. 2009]. A privacidade pode ser vista como um aspecto da confidencialidade. A confidencialidade define que uma informação não deve estar disponível ou divulgada a indivíduos, entidades ou processos não autorizados pela política de acesso [Shirey 2000]. Por sua vez, a privacidade é a proteção contra a exposição indevida de informações pessoais ou o desejo de controlar o nível de exposição e uso dessas informações.

2.4.2. Privacidade e anonimato

O anonimato é a qualidade ou condição do que é anônimo, isto é, sem identificação ou autenticação. Com o advento das mensagens por telecomunicações e, em particular, pela Internet, designa o ato de manter uma identidade escondida de terceiros [Wright 2004, Staples 2007]. O anonimato oferece algumas vantagens, como expressar opiniões polêmicas sem receio de represálias. Contudo, também pode ser usado de forma maliciosa, sendo por essa razão regrado pelas leis de vários países (como é o caso da Constituição brasileira, segundo a qual "É livre a manifestação do pensamento, sendo vedado o anonimato"). O valor do anonimato não reside na capacidade de ser anônimo, mas na possibilidade de atuação ou participação permanecendo inacessível a terceiros. [Staples 2007] comenta que indivíduos com capacidade de desvincular suas identidades de suas atividades têm melhores condições de controlar a divulgação de suas informações pessoais a terceiros.

Deve-se diferenciar o *anonimato total* do *anonimato parcial* (ou pseudo-anonimato) [Wright 2004]. No anonimato total, não é possível rastrear a origem da comunicação, por exemplo, uma carta sem assinatura e sem um endereço de retorno. Já o anonimato parcial faz uso de pseudônimos para mascarar uma identidade verdadeira, permitindo receber respostas sem a possibilidade de associar a origem da comunicação à identidade real. O anonimato parcial é uma alternativa a considerar quando o anonimato total não for permitido, por exemplo, em situações onde a identidade do usuário deva ser mantida oculta, mas este possa ser responsabilizado por seus atos. Todavia, o anonimato total na Internet é difícil de alcançar, pois a infraestrutura que mantém a Internet permite identificar as origens e destinos das comunicações.

O anonimato pode ser considerado uma ação extrema para manter a privacidade.

Sem divulgação alguma de informação pessoal, o controle sobre estas é completo, a não ser que elas sejam obtidas através de ações maliciosas. Todavia, ao empregar o anonimato, um indivíduo pode ter restrito seu acesso a serviços que exijam sua identificação, pode tornar-se menos acessível a comunicações iniciadas por outros indivíduos.

2.4.3. Informações privadas

Informação pessoal é um termo utilizado de forma genérica para identificar informações de diferentes indivíduos. Neste trabalho será utilizada a caracterização de informações privadas proposta por [Pearson et al. 2009], resumida a seguir:

- *Informações que Identificam o Indivíduo*: qualquer informação que pode ser usada para identificar ou localizar um indivíduo (nome ou endereço, por exemplo) ou informações que podem ser correlacionadas com outras informações para identificação do indivíduo (número de cartão de crédito ou de telefone).
- *Informações sensíveis*: informações sobre religião, raça, saúde, orientação sexual, partidária ou outras informações similares consideradas privadas. Estas informações exigem garantias adicionais, pois podem ser utilizadas para constranger o indivíduo. Outras informações consideradas sensíveis incluem dados financeiros ou sobre seu desempenho profissional.
- *Outras informações consideradas sensíveis que possam identificar o indivíduo*: tais como informações biométricas ou imagens de câmeras de vigilância em locais públicos.
- *Dados comportamentais*: dados coletados a partir do uso do computador ou outros dispositivos, como históricos de navegação na Internet ou contatos de amigos virtuais.
- *Dispositivos de identificação única*: outros tipos de informação que possam ser identificadas pelo uso de um dispositivo exclusivo do usuário, tais como endereço IP, dispositivos RFID ou *tokens* de gerência de identidade baseados em hardware.

A privacidade dessas informações possui três aspectos [Pfleeger e Pfleeger 2006]: sua *divulgação*, sua *sensibilidade* e as *partes afetadas*. Sob a ótica da divulgação, a privacidade pode ser definida de acordo com o nível de controle que se deseja dar a uma informação, ou seja, para quem o dono da informação deseja permitir o acesso. A partir do momento que um indivíduo divulga uma informação considerada privada a outro indivíduo (como um número de telefone celular), esta terá poder sobre a mesma, podendo divulgá-la a terceiros se o desejar.

Em relação à sensibilidade, é considerada uma informação sensível qualquer informação que possibilite uma violação de segurança, ou seja, uma violação da integridade, disponibilidade ou confidencialidade. Em [Stahl 2008] a sensibilidade da informação é exemplificada como dados pessoais sobre opiniões religiosas, políticas, condições de saúde ou origem étnica e portanto passíveis de proteção. A definição inequívoca do nível de sensibilidade de uma informação pessoal pode ser complexa. Por exemplo, para muitas

pessoas o valor de seus salários é um dado importante que poucos deveriam conhecer; já para outras pessoas é indiferente se seus colegas conhecem o valor de seu salário.

As entidades afetadas podem ser indivíduos, empresas, organizações, governos, etc. Todas as entidades atribuem níveis de sensibilidade às suas informações (ou às informações de seus elementos constituintes). Um hospital, por exemplo, considera as informações sobre seus pacientes como privadas [Beaver e Harold 2004], enquanto um governo considera privadas muitas informações militares ou diplomáticas.

2.4.4. Princípios da privacidade

Um dos primeiros trabalhos a discutir questões relativas à privacidade em ambiente computacional foi [Ware 1973], que definiu vários princípios (ou critérios a observar) relativos à privacidade de dados pessoais mantidos em um sistema informático. Mais recentemente, outros trabalhos retomaram a definição desses princípios de forma mais ampla, como [Fischer-Hübner 2001, Hinde 2003, Rezgui et al. 2003, Yee e Korba 2009]. Desses trabalhos é possível sintetizar o seguinte conjunto de princípios para a privacidade em um contexto mais amplo, não exclusivamente computacional:

1. *Responsabilidade*: uma organização é responsável pelas informações pessoais sob o seu controle e deve designar indivíduos responsáveis pela organização e conformidade dessas com a legislação e políticas internas. Devem existir termos sobre responsabilidade de uso, com sanções legais em caso do mau uso.

2. *Identificação de objetivos*: os objetivos para os quais as informações pessoais são coletadas devem ser identificados pela organização previamente ou enquanto a informação é coletada.

3. *Consentimento*: o consentimento dos indivíduos é necessário para a coleta, uso e/ou divulgação de informações pessoais. As informações não podem ser transferidas para outras entidades, salvo se autorizado e com um nível de proteção adequado.

4. *Limite de coleta*: a coleta de informação pessoal será limitada ao necessário para os fins identificados pela organização. As informações devem ser coletadas por meios conhecidos e com amparo legal.

5. *Limite de uso, divulgação e retenção*: as informações pessoais não devem ser utilizadas ou divulgadas para outros fins que não aqueles para os quais foram coletadas, exceto com o consentimento do indivíduo ou se exigido por lei. Além disso, as informações pessoais serão mantidas apenas o tempo necessário para o cumprimento desses propósitos.

6. *Precisão*: as informações pessoais deverão ser tão precisas, completas e atualizadas quanto for necessário para os propósitos definidos.

7. *Salvaguardas*: medidas de segurança adequadas à sensibilidade das informações devem ser usadas para protegê-las, tanto em relação à sua confidencialidade quanto à sua integridade.

8. *Transparência*: a organização deve tornar disponíveis aos indivíduos informações específicas sobre suas políticas e práticas relativas à gestão das informações privadas.

9. *Acesso individual*: a pedido, todo indivíduo deve ser informado da existência, uso e divulgação de suas informações pessoais e deve ser permitido o acesso a essa informação.

10. *Crítica à Conformidade*: um indivíduo deve ser capaz de criticar a precisão e integridade de suas informações e modificá-las se necessário. Também deve ser facultado ao indivíduo questionar os princípios anteriores ou os sujeitos responsáveis a respeito da política de privacidade adotada.

2.4.5. Privacidade em computação em nuvem

A privacidade é uma propriedade importante para a computação em nuvem, seja em termos de conformidade legal ou confiança do consumidor, a ponto de ser citada no relatório anual da Ernst & Young [Ernst & Young 2010] como um dos tópicos mais importantes para o ano de 2010. Logo, a proteção da privacidade é uma questão-chave e precisa ser considerada em todas as fases de um projeto para esse ambiente. De forma geral, todos os aspectos apresentados na seção 2.4.3 são levantados nas pesquisas sobre privacidade na nuvem. Ainda assim, vários questionamentos sobre privacidade surgem, na medida em que essa tecnologia evolui [Mather et al. 2009]:

- *Acesso*: os donos dos dados têm o direito de saber quais informações são mantidas e, em alguns casos, de solicitar a remoção dessas informações. Se um usuário exerce o seu direito de solicitar ao provedor a eliminação dos seus dados, será possível garantir que todas as suas informações foram eliminadas da nuvem?

- *Aderência*: quais são os requisitos de conformidade à privacidade neste ambiente? Quais leis, regulamentos, normas ou compromissos contratuais regem o uso das informações, e quem são os respectivos responsáveis? Um ambiente de nuvem pode atravessar várias jurisdições, no caso de dados armazenados em vários países. Qual o foro competente para regular esse ambiente ou as informações armazenadas nele?

- *Armazenamento*: em qual parte da nuvem as informações estão armazenadas? Estão em um centro de dados de outro país? A legislação sobre privacidade pode limitar a capacidade dos provedores em transferir alguns tipos de informações para outros países. Todavia, quando os dados são armazenados na nuvem, essa transferência pode ocorrer sem o conhecimento dos mesmos.

- *Retenção*: por quanto tempo a informação pode ser retida na nuvem? Quais as políticas de retenção e descarte dessas informações? Quem rege essas políticas, o consumidor que armazenou as informações ou o provedor de computação em nuvem? Quais as exceções à política?

- *Destruição*: como deve ocorrer a destruição das informações que identificam o consumidor? Como garantir que não foram conservadas cópias destas? A disponibilidade através de replicação pode ser um problema no momento da destruição

de informações. Será que as réplicas da informação foram destruídas ou apenas tornaram-se inacessíveis?

- *Auditoria*: como as organizações consumidoras podem monitorar e verificar se seus fornecedores estão cumprindo os requisitos de privacidade?
- *Violação da privacidade*: em casos confirmados de violação, quais os responsáveis pela notificação, processos (e custos associados)? Como determinar a culpa de cada entidade envolvida?

Os riscos e ameaças à privacidade diferem de acordo com cenário de uso da computação em nuvem. Para [Pearson et al. 2009], existem alguns cenários de riscos que podem afetar a privacidade: o usuário de serviços em nuvem poderia ser forçado ou persuadido a aceitar o monitoramento das suas atividades ou fornecer informações pessoais contra sua a vontade, ou de alguma outra forma na qual não se sinta confortável; este poderia ver suas informações armazenadas na nuvem perdidas ou divulgadas; os provedores de serviços poderiam utilizar as informações pessoais para outros fins que não os definidos inicialmente. Não existem só riscos para os usuários: em caso de problemas, o implementador do PaaS poderia ser responsabilizado legalmente pela exposição de informações sensíveis, ocasionando prejuízos financeiros e perda de credibilidade.

As normas e recomendações existentes sobre segurança, tais como a *IT Infrastructure Library* (ITIL) [ITIL 2010] ou o padrão ISO 27001:2005 [ISO 2005], não foram criadas pensando nos ambientes virtuais de armazenamento de dados e fornecimento de serviços de nuvem. Neste contexto, o trabalho [Doelitzscher et al. 2010] identifica os seguintes problemas na computação em nuvem:

- *Cumprimento das leis e políticas*: os consumidores são responsáveis pela segurança e integridade de seus dados, inclusive nos casos em que estes estão sendo sob a custódia de terceiros.
- *Privilégios de controle de acesso*: o armazenamento ou tratamento de informações sensíveis na nuvem gera um risco adicional: os serviços de computação em nuvem são controlados por terceiros. É necessário um controle de acesso restrito sobre os administradores para evitar acesso indevido às informações dos consumidores.
- *Segmentação das Informações*: as informações de vários consumidores da nuvem podem ser armazenadas no mesmo disco rígido físico, separadas pelo uso da virtualização. Para a proteção das informações contra acesso não-autorizado, os provedores de serviços na nuvem precisam prover criptografia dos dados. Além disso, é necessário manter registros de auditoria, pois falhas durante o processamento das informações podem acarretar em perda destas.
- *Incidentes de segurança*: os provedores de serviços na nuvem oferecem poucos recursos no caso de incidentes de segurança. Em um ambiente onde máquinas virtuais são continuamente iniciadas e encerradas por vários consumidores, a atualização das informações dos usuários e dos registros de atividade é uma tarefa desafiadora. Normalmente, apenas o registro de atividades de um usuário na nuvem é oferecido em casos de investigação de incidentes.

Alguns dos problemas levantados podem ser resolvidos com ferramentas tradicionais de segurança, como gerência de controle de acesso, ferramentas de auditoria, gerência de identidade, etc. Todavia, ainda restam algumas lacunas relativas à proteção da privacidade. A próxima seção apresenta diversos trabalhos que visam a proteção da privacidade dos usuários em ambientes de computação em nuvem.

2.4.6. Abordagens de proteção da privacidade

Em geral, a proteção da privacidade de dados pode ser realizada através de leis de proteção à privacidade definidas por cada governo, da auto-regulamentação para práticas leais e códigos de conduta promovidas por entidades ao manipular informações, por tecnologias que aumentem a privacidade, e também através da educação sobre privacidade a usuários e profissionais de TI. Infelizmente, a privacidade é muitas vezes mal gerida e, por consequência, ocorrem abusos no uso das informações. Exemplos bem conhecidos são a divulgação de informações pessoais a terceiros sem o consentimento explícito de seus proprietários legítimos, a construção de perfis de usuários a partir do relacionamento de informações heterogêneas [De Capitani di Vimercati e Samarati 2006].

Do ponto de vista tecnológico, existem diversas abordagens que permitem melhorar a proteção da privacidade das informações. Já nos anos 1970, o trabalho [Turn e Ware 1975] sugeria algumas medidas para a proteção de dados pessoais em sistemas computacionais, como reduzir a exposição de informações, limitando a quantidade de dados armazenada e usando amostras aleatórias em vez de coletas completas; reduzir a sensibilidade dos dados ou adicionar erros sutis a eles; modificar ou retirar alguns dados para torná-los anônimos (mascarar o nome, por exemplo); e também criptografar os dados.

Os requisitos de privacidade variam de acordo com as leis de cada país ou a vontade do consumidor. Assim, é importante que os provedores que compartilham e trocam informações adotem e apliquem diretivas de privacidade. O perfil XSPA (*Security and Privacy Authorization* [OASIS 2009a] – uma parte da especificação XACML [OASIS 2005b]) auxilia as entidades na troca e definição de requisitos de privacidade nestes casos.

Na sequência do texto serão discutidos alguns trabalhos específicos recentes para a proteção da privacidade em ambientes de computação em nuvem.

2.4.6.1. Gerência de identidades digitais

Normalmente, usuários novos têm de estabelecer sua identidade antes de utilizar um serviço na nuvem. Isto se dá, geralmente, pelo preenchimento de um formulário *online* solicitando informações sensíveis (nome, endereço, número do cartão de crédito, por exemplo). Este processo deixa rastros de informações pessoais que, se não forem devidamente protegidas, podem ser roubadas. Neste contexto, o trabalho [Bertino et al. 2009] propõe a criação de um serviço para a gestão de identidades digitais (*IdM – Digital Identity Management*) para minimizar o risco de roubo de identidade e fraude.

Os serviços oferecidos na nuvem são heterogêneos e podem utilizar atributos distintos para a identificação dos usuários. Assim, surgem problemas de interoperabilidade que vão desde o uso de *tokens* de identidade diferentes, como os certificados X.509 ou o

uso de informações diferenciadas para identificar o usuário (CPF ou e-mail, por exemplo). O uso de informações diferentes para montar uma identidade cria outro problema: a heterogeneidade de identidades, que ocorre quando usuários e provedores de serviços usam vocabulários diferentes para os atributos de uma identidade. Esse conjunto de identidades diferentes dificulta a utilização da nuvem, pois o usuário pode fornecer informações desnecessárias ou mesmo erradas para um provedor.

Neste contexto, o uso de um gerenciador de identidades para identificar o usuário dificultaria a violação da privacidade do mesmo. A fim de solucionar o problema da gestão de identidades heterogêneas, os autores propõem a utilização do *IdM* combinado com um protocolo de privacidade (utilizando técnicas de criptografia, tabelas de correspondência, dicionários e ontologias), a fim de identificar o usuário nos vários serviços da nuvem. Esse protocolo utiliza um conjunto de provas de conhecimento (*Aggregate Zero Knowledge Proofs of Knowledge - AgZKPK*) que permite ao cliente comprovar seus atributos de identificação sem a necessidade de informá-los de forma explícita.

[Bertino et al. 2009] consideram a adoção do *IdM* em conjunto com outras entidades: Provedores de Identidade (*IdP - Identity Providers*), provedores de serviços na nuvem (*CSP - Cloud Service Providers*), sistemas de registro (*Registrars*) e usuários. O *CSP* provê acesso aos dados e ao software disponível; os *IdPs* fornecem os atributos para a identificação do usuário e o controle no compartilhamento de informações. Nessa proposta, os sistemas de registro são elementos adicionais que armazenam e gerenciam as informações relacionadas à identificação dos atributos utilizados, sendo que cada registrador contém tuplas de atributos de identificação do usuário. As informações armazenadas nos registradores não incluem valores de identificação não-cifrados.

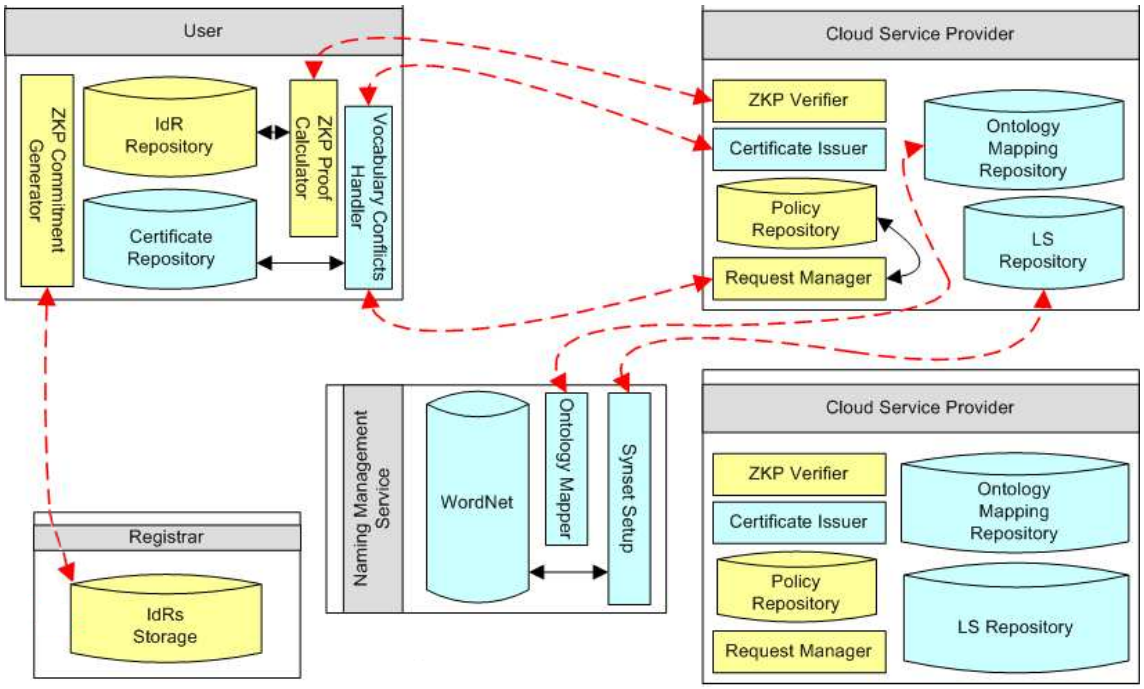

Figura 2.6. Sistema de gerência de identidades digitais [Bertino et al. 2009].

A arquitetura proposta (Figura 2.6) é composta por quatro componentes: o sistema

de registro, o provedor de serviços, o agente do usuário e o gerenciador de identidades heterogêneas. O sistema de registro é o responsável pelo registro de identidades no cliente e fornece mecanismos para recuperação de dados públicos utilizados no protocolo AgZKPK. O agente do usuário é composto por três módulos: gerador de compromissos ZKP[1], calculador da prova ZKP e gerenciador de conflitos de vocabulários. O gerador de compromissos fornece funções para o cálculo de compromissos de Pedersen (*Pedersen commitments* [Pedersen 1992]) sobre os atributos da identidade, o calculador gera a prova AgZKPK a ser fornecida ao CSP. O gerenciador de conflitos controla os nomes de atributos que correspondem aos parâmetros a ser enviados ao CSP e gerencia os certificados de prova de identidade armazenados localmente.

O provedor de serviços é composto por quatro módulos: um gerenciador de pedidos, um gerenciador de mapeamento de caminhos, o certificado do emissor e verificador ZKP e três repositórios responsáveis pelo armazenamento do mapeamento de ontologias, dos conjuntos de sinônimos e das políticas de verificação de identidade. O gerenciador de pedidos manipula as solicitações dos clientes e verifica junto a eles os atributos de identificação necessários para a verificação. O verificador ZPK testa a prova AgZKPK. Finalmente, o serviço de gerência de heterogeneidade provê várias funções que são compartilhadas por todos os CSPs e consiste em dois módulos: *Synset SetUp* e gerenciador de ontologias. O módulo *Synset SetUp* retorna, consultando um dicionário local, um conjunto de sinônimos para um determinado termo e o gerenciador de ontologias fornece as funcionalidades para realizar o mapeamento entre duas ontologias.

2.4.6.2. Gerenciador de privacidade

O artigo [Pearson et al. 2009] propõe um gerenciador de privacidade que usa técnicas de ofuscação de dados (uma técnica na qual algumas características dos dados originais permanecem presentes nos dados cifrados). O método utiliza uma chave de ofuscação fornecida pelo usuário e conhecida do gerenciador, mas que não é fornecida ao provedor de serviços na nuvem. Desta forma, os autores acreditam que não é possível recuperar as informações nem roubar os dados dos usuários. O provedor do serviço tem responsabilidades legais ao tratar informações abertas dos usuários, mas os dados ofuscados não permitem a identificação do indivíduo. Assim, o provedor do serviço não está sujeito às sanções legais no tratamento dos dados ofuscados.

Entretanto, para as aplicações em nuvem não é prático trabalhar somente com dados ofuscados. Neste caso, o gerenciador de privacidade acrescenta duas características: *preferências* e *personalidade*, que possibilitam aos usuários comunicarem aos provedores de serviços como eles desejam que suas informações sejam tratadas (e que indiretamente constituem um consentimento do usuário para o uso das informações pelo provedor). As preferências indicam quais e como os dados podem ser visualizados, enquanto a personalidade possibilita ao usuário personificar diferentes visões da mesma identidade (anônimo, visão parcial, visão total).

[1]ZPK (*zero-knowledge proof*) é um método interativo que permite a uma parte provar para outra que uma determinada declaração é verdadeira sem revelar qualquer informação, a não ser a veracidade da declaração. Normalmente a prova ocorre pela utilização de fórmulas matemáticas.

A proposta sugere o uso de um Módulo de Plataforma Confiável (*Trusted Platform Module* - TPM), um componente de hardware que provê serviços de criptografia, geração e verificação de chaves digitais, cálculo e verificação de *hash* de informações e armazenamento de informações para verificação de identidades [Trusted Computing Group 2010]. O TPM é um módulo inviolável e pode fornecer vários serviços de segurança necessários para soluções na nuvem.

Os autores descrevem três arquiteturas possíveis para o gerenciador de identidade:

- *Gerenciador de privacidade no cliente*: o gerenciador de privacidade executa no lado do cliente e auxilia os usuários na proteção de sua privacidade ao acessar serviços fornecidos pela nuvem (Figura 2.7). O principal componente do gerenciador é provido pelo *serviço de ofuscação e desofuscação*, que reduz a quantidade de informações sensíveis fornecidas para a nuvem. Nesta arquitetura, o TPM é utilizado para proteger as chaves de ofuscação no cliente, garantir a integridade do gerenciador de privacidade e as identidades utilizadas pelo usuário. Como a proposta é baseada no TPM, as informações não estarão disponíveis para acesso mesmo com a violação deste, pois neste caso ocorrerá a perda da chave de ofuscação sob sua responsabilidade.

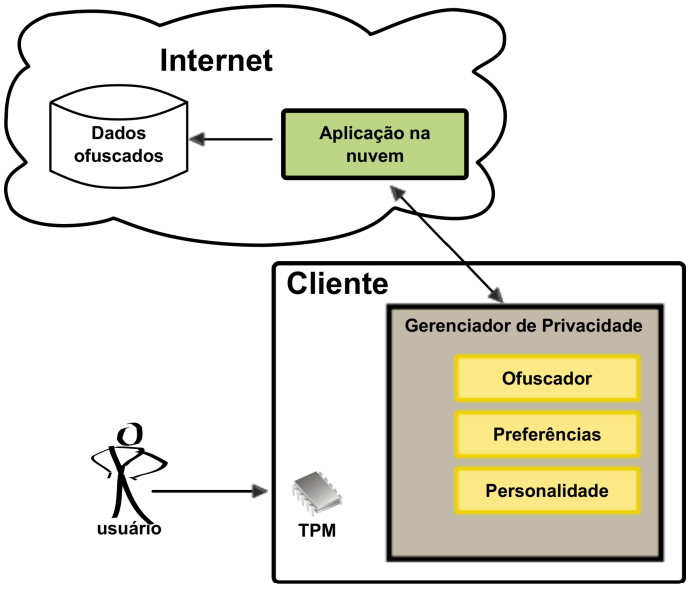

Figura 2.7. Gerenciador de privacidade no cliente [Pearson et al. 2009].

- *Gerenciador de privacidade em uma nuvem híbrida*: uma alternativa à gerência de privacidade no cliente é a possibilidade de gerenciar a privacidade entre nuvens distintas. Esta solução é indicada para ambientes corporativos, nos quais exista uma proteção adequada; sua aplicação principal seria o controle de informações pessoais circulando da nuvem privada para a nuvem pública. A proposta sugere que o gerenciador e o TPM possam executar em ambientes virtualizados [Dalton et al. 2009], com o uso combinado de *proxies* e outras ferramentas tradicionais de segurança. Os autores alertam para o problema da escalabilidade da proposta, que neste caso pode ser afetada em função do nível de tráfego entre as nuvens.

- *Gerenciador de privacidade para infomediários na nuvem*: neste caso, o gerenciador de privacidade atua entre dois infomediários[2]. Ele age em nome do usuário e decide o grau permitido de transferência de informações. O nível de privacidade é baseado nas políticas de transferência de informações especificadas pelo usuário, no contexto do serviço e, se possível, em uma avaliação da confiabilidade do ambiente prestador de serviços. O infomediário poderia ser uma associação de consumidores ou outra entidade qualquer na qual o usuário confie.

Os autores sugerem como exemplo de uso um cenário com um fotógrafo portando uma câmera com GPS embutido, que acrescenta informações geográficas à imagem. Neste cenário, o fotógrafo deseja compartilhar suas fotos com alguns familiares, mas algumas destas também podem ser comercializadas. Para compartilhar apenas com os familiares, basta definir essa intenção nas preferências. Para a comercialização, o fotógrafo deseja retirar as informações de GPS para impedir que outros concorrentes possam tirar fotos idênticas do mesmo local. Neste caso, o gerenciador é utilizado para ofuscar as informações geográficas antes do compartilhamento.

2.4.6.3. Privacidade como um serviço

O trabalho [Itani et al. 2009] apresenta uma proposta para assegurar a privacidade das informações pessoais dos usuários na nuvem, observando a legislação correspondente. Conhecido como *Privacy as a Service* (PasS), o projeto é baseado em um conjunto de protocolos de segurança, com processamento e auditoria das informações sensíveis através de processadores criptográficos. Os processadores criptográficos são indicados por serem invioláveis e à prova de falsificação (de forma análoga ao TPM), de acordo com as especificações definidas em [FIPS 140-2 2001]. Esses processadores são posicionados em um domínio de execução seguro e confiável dentro da infraestrutura da nuvem, devendo estar física e logicamente protegidos contra acessos não-autorizados.

O modelo proposto é baseado em uma solução de nuvem típica composta de duas partes: um *provedor*, que gerencia e opera uma infraestrutura de nuvem para armazenamento e serviços, e um *usuário*, que utiliza o armazenamento na nuvem e os recursos remotos para processamento de dados. O projeto oferece ao usuário o controle completo sobre os mecanismos de privacidade aplicados às informações na nuvem. O serviço de privacidade proposto baseia-se no grau de sensibilidade das informações do usuário para definir os níveis de confiança no provedor do serviço. O serviço de privacidade oferece suporte a três níveis de confiança:

1. *Confiança total*: este nível se aplica às informações não-sensíveis, que podem ser processadas e armazenadas na nuvem sem uso de criptografia. O provedor é considerado totalmente confiável para o armazenamento e processamento de informações deste nível.

[2]Empresas infomediárias são intermediárias de informações, cujo negócio é pesquisar e analisar informação, desenvolvendo análises detalhadas do mercado e caracterização dos clientes para utilização por outras entidades. São especializadas em veiculação de conteúdo via internet.

2. *Confiança parcial*: este nível envolve as informações que precisam ser armazenadas cifradas, seja por questões jurídicas ou por regras de conformidade (como registros médicos ou de transações financeiras, por exemplo). Neste nível o consumidor confia no provedor para armazenar suas informações cifradas utilizado chaves fornecidas por ele.

3. *Sem confiança*: este nível aplica-se a informações sensíveis que devem ser ocultadas do provedor. Este tipo de informação deve ser armazenado cifrado, usando chaves criptográficas especificadas pelo usuário e transformadas em repositórios isolados na nuvem. Esses repositórios são configurados, distribuídos e mantidos por um terceiro confiável compartilhado pelo provedor e pelo usuário. A figura 2.8 apresenta o modelo deste sistema e a interações entre o provedor, o usuário e o terceiro confiável.

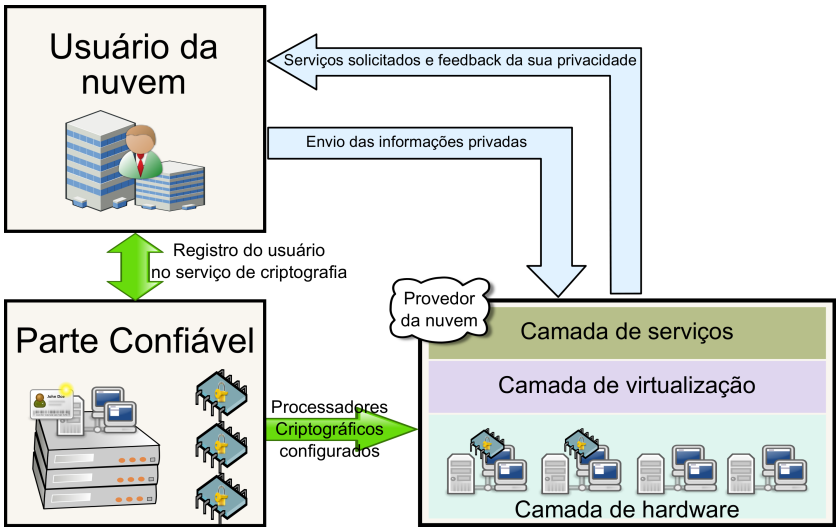

Figura 2.8. Modelo do serviço de privacidade *PasS* (adaptado de [Itani et al. 2009]).

Como pode ser observado na figura 2.8, a proposta faz uso combinado de processadores criptográficos e protocolos de privacidade. Os processadores são posicionados em um terceiro confiável (*trusted third party*) e acessível ao usuário e ao provedor. O uso de uma camada de virtualização é indicado para permitir o compartilhamento dos processadores entre vários usuários simultâneos. A solução prevê a criação de uma camada de software para separar os componentes do serviço a ser executados na nuvem, de acordo com os níveis de confiança definidos e suportados pelo modelo. Finalmente, o modelo especifica um protocolo de *feedback* sobre a privacidade obtida. O objetivo desse protocolo é informar o usuário sobre os níveis de privacidade de suas informações e alertar sobre a possibilidade de vazamento de dados ou outras situações de risco que poderiam comprometer a privacidade de suas informações.

2.4.6.4. Serviços de nuvem em conformidade com a legislação de privacidade

A fim de aderir à legislação governamental alemã sobre privacidade, o trabalho [Doelitzscher et al. 2010] propõe um modelo em camadas, denominado *CloudDataSec*,

para ser implementado em projetos de infraestrutura de nuvens. Este modelo é composto das seguintes camadas (figura 2.9):

- *Análise de risco*: estabelece e gerencia as avaliações de riscos sobre a terceirização de serviços na nuvem, auxiliando na identificação das informações e serviços que devem permanecer dentro dos limites da organização consumidora.
- *Orientações de segurança*: descreve as políticas e restrições legais relativas à privacidade aplicáveis ao ambiente de computação em nuvem.
- *Monitoração de qualidade de serviço (QoS)*: um acordo de nível de serviço (*Service Level Agreement* - SLA) entre clientes e fornecedores especifica os níveis de exigência de segurança e privacidade, além de garantir a segurança jurídica dos contratos sobre os serviços.
- *Criptografia dos dados e registros (logs)*: a criptografia visa proteger a confidencialidade e integridade das informações, enquanto os registros fornecem um histórico completo das atividades do usuário.
- *Comunicação criptografada*: nesta camada são utilizados os protocolos padronizados como SSH, IPSec e suas implementações.

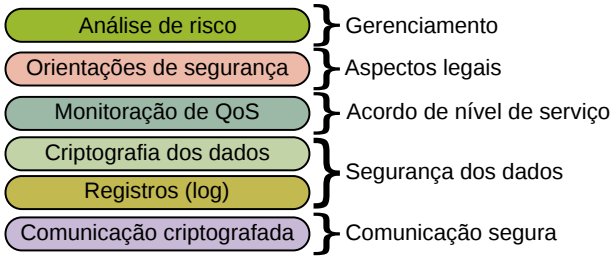

Figura 2.9. Camadas de segurança e privacidade para nuvem [Doelitzscher et al. 2010].

Para os autores, a aplicação do modelo *CloudDataSec* garante a aderência à legislação e a proteção das informações contra alguns ataques, como *man-in-the-middle* (ataque do intermediário). Além de propor um modelo de aderência, o trabalho sugere três níveis de garantia de segurança, sumarizados na tabela 2.1. Para ilustrar a aplicabilidade da proposta, o trabalho descreve um cenário onde uma empresa de desenvolvimento deseja terceirizar serviços na nuvem. Após uma análise de riscos, os seguintes níveis de segurança são identificados:

- *Nível Básico* para o desenvolvimento das páginas Web da empresa; não existem informações sensíveis armazenadas no servidor Web de desenvolvimento.
- *Nível Avançado* para hospedar o site da empresa. Não há restrição sobre o número de máquinas virtuais na mesma máquina física (*host*), mas somente máquinas virtuais desse domínio devem ser permitidas no mesmo *host*. No caso de um incidente de segurança, a máquina virtual comprometida é movida para a quarentena, e uma imagem íntegra da mesma é lançada, para propiciar alta disponibilidade ao serviço.

Tabela 2.1. Níveis de segurança para serviços na nuvem [Doelitzscher et al. 2010]

Serviço	Básico	Avançado	*Premium*
Local da VM	*pool aberto*	*pool* restrito	*host* privado
Identificação	e-mail, cartão	documento	terceiro confiável
Firewall	√	√	√
Administração	*GUI firewall*	*GUI firewall*	*GUI firewall*
Monitor de protocolo	–	√	√
Firewall de aplicação	–	√	√
Quarentena para sistemas comprometidos	–	√	√
Reinício do serviço	–	VMs seguras (até 3)	VMs seguras
Acesso ao sistema em quarentena	–	SSH	SSH, terminal

- *Nível Premium* para hospedar uma loja *on-line*; neste caso, um vazamento de dados dos usuários pode afetar a reputação da empresa. Em caso de incidente, o sistema é movido para a quarentena para evitar o vazamento de informações. O serviço não é relançado automaticamente, para evitar a repetição do ataque e a possibilidade de exposição de dados pessoais.

As implementações descritas nos cenários envolvem o desenvolvimento de um conjunto de módulos, entre os quais o Serviço de Gestão e Monitoramento de Segurança (*Security Management and Monitoring as a Service - SMaaS*). A Figura 2.10 ilustra a arquitetura do *SMaaS* em um ambiente de nuvem simplificado. Por sua vez, o módulo *SMaaS* é constituído dos seguintes componentes: Criptografia, Infraestrutura de Chave Pública (PKI), monitoramento de SLA, módulo de verificação de políticas (configuração, acesso, estado do sistema, etc), prevenção de vazamento dos dados (*Data Leakage Prevention* - DLP), registros (log) e um sistema de detecção de intrusão (IDS).

2.4.6.5. Preservação da privacidade em computação em nuvem

Existem muitos provedores de serviços na nuvem, possibilitando aos usuários acessar serviços diversos, mas quando as informações são trocadas entre os serviços, surge o problema da divulgação das informações e da consequente violação da privacidade [Wang et al. 2009]. A partir dessa afirmação, os autores propõem um novo algoritmo de anonimato a ser aplicados nos micro-dados (partes da informação) antes que estes sejam publicados na nuvem. Segundo os autores, o uso da criptografia não seria suficiente para garantir a privacidade das informações, já que o provedor do serviço precisa decifrar as informações antes do processamento.

Os autores indicam a utilização de uma base de conhecimento externa, não necessariamente armazenada na nuvem, como registros públicos ou outros canais na Internet. A junção dos dados anonimizados com a base externa possibilitaria ao provedor do serviço realizar pesquisas e obter os dados necessários para resolver alguns problemas.

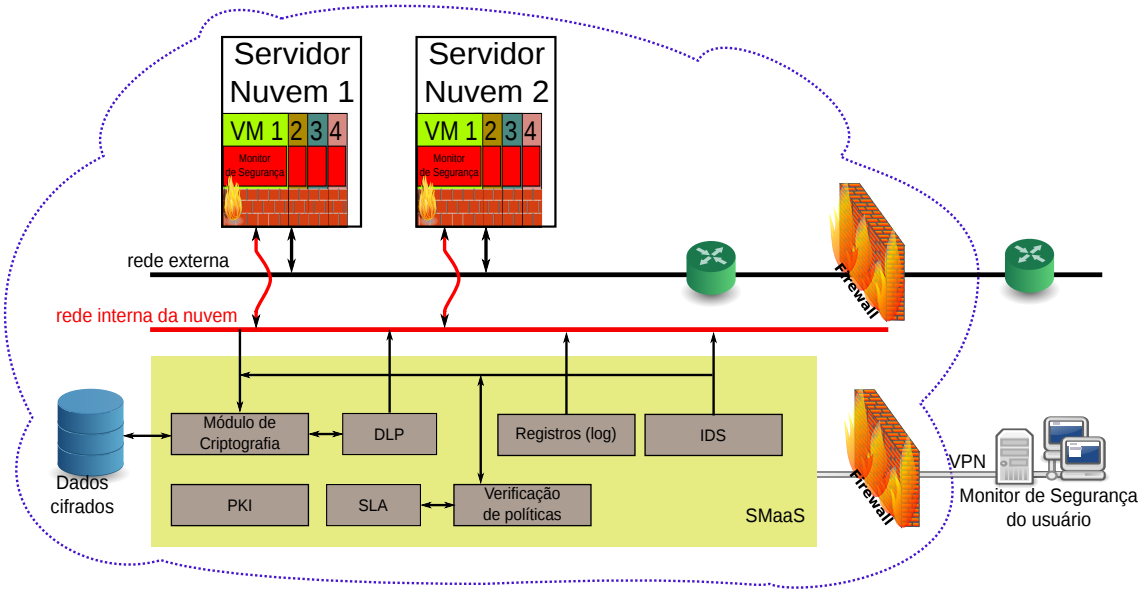

Figura 2.10. *SMaaS* em um ambiente de nuvem [Doelitzscher et al. 2010].

No trabalho os autores utilizam um cenário de acidentes de trânsito, onde informações como cores do veículo são suprimidas e profissão específica é substituída pela grande área (por exemplo: a informação "dentista" é substituída por "área médica", e "diretor de escola" seria substituída por "área de ensino"). Com este exemplo, os autores querem mostrar que é possível montar relatórios estatísticos confiáveis, mesmo sem acesso às informações completas.

2.4.6.6. Privacidade e Criptografia

A circulação e manutenção de dados pessoais na nuvem acaba por exigir o uso da criptografia. Todavia, esta impõe limites estritos para a utilização dos dados ao provedor de serviços, por exemplo, se os dados são armazenado em texto não-cifrado, pode-se procurar um documento especificando uma palavra-chave, o que deixa de ser possível com textos cifrados usando algoritmos tradicionais [Chow et al. 2009]. Contudo, pesquisas recentes em criptografia possibilitaram realizar algumas operações em dados cifrados, como indexação ou pesquisas [Song et al. 2000, Boneh e Waters 2006, Shen et al. 2009, Singh et al. 2010].

Outra possibilidade promissora consiste no uso de algoritmos de criptografia totalmente homomórfica (*fully homomorphic encryption*). Esses algoritmos permitem a aplicação de funções arbitrárias sobre dados criptografados, sem a necessidade de decifrá-los. Formalmente, considerando um algoritmo de criptografia totalmente homomórfica c, dados em claro $a_1, \ldots, a_n$ e seus respectivos dados cifrados $c_1 = c(a_1), \ldots, c_n = c(a_n)$, pode-se rapidamente computar $c(f(a_1, \ldots, a_n))$ a partir de $f(c_1), \ldots, f(c_n)$, para qualquer função computável f. Apesar da noção de cifragem totalmente homomórfica ter sido proposta por Rivest, Adleman e Dertouzos em 1978, só recentemente foram concebidos algoritmos que se enquadram nesta categoria [Gentry 2009]. A possibilidade de realizar computações arbitrárias sobre dados cifrados sem revelar seu conteúdo abre

uma série de novas oportunidades para a proteção da privacidade.

2.5. Problemas em aberto

A segurança das informações é um dos assuntos mais questionados quando está em discussão a utilização ou a migração dos sistemas tradicionais para a nuvem computacional. Os assuntos discutidos sempre recaem em algum aspecto relacionado à segurança, inclusive tópicos a princípio pouco relacionados com segurança, como desempenho [Somani e Chaudhary 2009]. Porém, se um sistema tem problemas de desempenho podem ocorrer condições de corrida ou violações por inconsistência nas políticas. Se a alteração de uma política não foi tornada efetiva devido à lentidão de atualização, então a demora na correção do problema pode causar perdas irreparáveis.

A computação em nuvem traz redução de gastos com recursos locais, por exemplo, porém podem haver riscos associados a essa vantagem porque é necessária a delegação do controle sobre os dados para entidades terceirizadas [Cachin et al. 2009]. Na verdade surge ai a necessidade forte de confiança (*trust*) do consumidor no provedor e vice-versa, pois se o provedor armazena dados cifrados, pode estar fornecendo um repositório para material criminoso, por exemplo, por outro lado se o consumidor não cifra seus dados pode ter sua privacidade (Seção 2.4) violada intencional ou acidentalmente [Birman et al. 2009]. Adicionalmente, países e regiões possuem suas próprias leis regrando a localização física do armazenamento dos dados e suas formas de proteção (e.g. *USA Patriot Act*).

A disponibilidade é uma das preocupações em serviços online, pois podem ocorrer períodos de inatividade ocasionados por falhas de comunicação, de softwares ou por ataques. O armazenamento de dados em locais remotos necessita de garantias de integridade fornecidas pelo provedor da nuvem. O mal funcionamento de um programa no provedor pode liberar o acesso aos dados privados do consumidor, então estes podem ser alterados os copiados. Empresas tradicionais têm receio de armazenar dados fora de seu domínio, porque uma vez que o dado tenha sido comprometido (perdido, danificado ou roubado) nenhum SLA ou contrato poderá reparar tal perda. Existe a preocupação com casos onde um serviço precisa estar conectado simultaneamente a diferentes provedores de nuvem para funcionar sendo que se um destes parar de operar – dependendo do nível de acoplamento funcional entre os provedores, diferentes tipos de clientes podem ser prejudicados.

Quando vários usuários utilizam um mesmo provedor de maneira colaborativa, a consistência durante o acesso concorrente aos recursos também deve ser garantida. Um mesmo sistema de arquivos pode estar sendo acessado por sistemas operacionais diferentes (máquinas virtuais), como se fosse um *storage* (para o qual não há um hardware e um servidor específico para a mediação dos acessos). Em computação em nuvem, se não houver um *data center* para o armazenamento de dados quem terá que fazer o papel de servidor de *storage* é o hipervisor e este não está preparado para tal função. Os desafios são muitos, envolvendo grandes áreas de estudo – abrangendo as diferentes camadas e modelos de implantação de computação em nuvem [Birman et al. 2009].

A grande maioria das APIs de armazenamento ainda são proprietárias, ou não foram totalmente padronizadas, assim os consumidores não podem extrair facilmente seus dados e programas de um domínio para outro. A padronização das APIs permitirá que desenvolvedores implantem serviços com portabilidade ou compatibilidade para

armazenamento de dados em vários provedores de computação em nuvem. Com esta abordagem é possível desenvolver técnicas de replicação ou *backup* para não depender integralmente de um único provedor, pois se este falhar o consumidor poderá acionar seu plano de contingência, recuperando cópias dos dados de outro provedor [Wood et al. 2010]. A padronização de APIs também permitirá que os mesmos softwares sejam utilizados em uma nuvem privada e pública – modelo híbrido (e.g. *Eucalyptus*) – o que mudaria seria apenas a origem e destino (locais/remotos) dos dados, por exemplo.

Atualmente, cada nuvem possui sua própria solução para o gerenciamento de identidades. Várias tentativas estão sendo propostas e adaptadas, porém ainda não existe nenhuma padronização ou abordagem que ofereça fácil portabilidade. A crescente demanda por segurança e conformidade com leis e políticas também pode ser uma boa razão para a utilização de computação em nuvem, pois somente uma entidade – o provedor de computação em nuvem – deverá ser capaz de arcar com os custos para fornecer elevados níveis de segurança e auditabilidade para um grande número de consumidores.

Requisitos de auditabilidade (e.g. *Human Services Health Insurance Portability and Accountability Act* – HIPAA) devem ser fornecidos para que dados corporativos possam ser movidos para nuvens computacionais, permitindo que o acesso aos mesmos possa ser rastreado [Wang et al. 2010a]. Empresas tradicionais enfrentam ameaças/ataques internos e externos, porém, na nuvem computacional as responsabilidades são multi-parte, ou seja, aos consumidores de serviços de nuvem cabe a preocupação com a segurança em nível de aplicação; ao provedor cabe prover segurança física, em nível de *firewall*, IaaS, sistema de arquivos, isolamento de VMs etc.; a entidades terceirizadas contratadas pelo consumidor (como o serviço de identidade, por exemplo) cabe gerenciar configurações e dados sensíveis (incluindo a privacidade).

O usuário da nuvem deve ter meios para se proteger de seu provedor, pois é este que controla a camada de software subjacente. O consumidor deve utilizar contratos com cláusulas claras e em conformidade com as leis do país onde a prestação do serviço de computação em nuvem está sendo oferecida. A auditoria deveria ser acessível como uma camada adicional, fora do domínio do administrador de sistema operacional virtualizado – ficando muito mais segura do que se fosse embutida na aplicação [Wang et al. 2010a].

2.6. Considerações finais

Ao longo deste trabalho, foram discutidas vantagens e limitações que da computação em nuvem. Nós entendemos que a computação em nuvem promete revolucionar o paradigma da computação principalmente por transferir para um provedor - especializado na prestação de serviço e infraestrutura computacional - uma responsabilidade que o consumidor precisa assumir mesmo não estando preparado para isto antes da mesma. Porém, como qualquer prestação de serviço a computação em nuvem, por ser recente, carece de credibilidade, pois aspectos de segurança e privacidade são sempre uma preocupação pelo danos que podem causar as vítimas de violações que possam causar aos mesmos.

O ambiente de computação em nuvem traz uma série de desafios à privacidade, por exemplo: como limitar a coleta das informações? Como garantir o uso correto das informações a partir do momento que estas foram armazenadas na nuvem? Como garantir a destruição das informações na nuvem? A transição da computação pessoal para a

computação em nuvem fornece uma nova gama de aplicações e possibilidades de usos, mas também implica em novos problemas e desafios. A inadequação dos mecanismos convencionais de gestão de privacidade e a falta de ferramentas tecnológicas adequadas para permitir aos usuários controlar a exposição de suas informações sensíveis são limitações que limitam o uso desse ambiente.

Na prática, mesmo antes da computação o ser humano sempre teve que lidar com problemas de segurança e privacidade, a diferença sempre foi o contexto e os mecanismo utilizados para tal. Assim, acreditamos que temas como confiança (*trust*) e privacidade ganhem ainda mais importância porque não se espera, como comentado no texto, que um sujeito que não confia em um banco deposite seu dinheiro lá. Ainda, não se espera que um sujeito que não confia na capacidade de guardar sigilo de um contabilista entregue suas informações privadas ao mesmo. Mas, se espera que a computação em nuvem traga muitas opções de escolha para que os provedores que oferecem o melhor conjunto de propriedades de segurança e de privacidade sejam os mais demandados e possam melhorar cada vez seus serviços, até atingirem níveis satisfatórios de prestação de serviço.

Referências

[Armbrust et al. 2010] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., e Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4):50–58.

[Beaver e Harold 2004] Beaver, K. e Harold, R. (2004). *The Practical Guide to HIPAA Privacy and Security Compliance*. Auerbach Publications.

[Bertino et al. 2009] Bertino, E., Paci, F., Ferrini, R., e Shang, N. (2009). Privacy-preserving digital identity management for cloud computing. *IEEE Data Engineering Bulletin*, 32(1):21–27.

[Bhattacharjee 2009] Bhattacharjee, R. (2009). *An analysis of the cloud computing platform*. MSc thesis, System Design and Management Program, Massachusetts Institute of Technology.

[Birman et al. 2009] Birman, K., Chockler, G., e van Renesse, R. (2009). Toward a cloud computing research agenda. *ACM SIGACT News*, 40(2):68–80.

[Boneh e Waters 2006] Boneh, D. e Waters, B. (2006). Conjunctive, subset, and range queries on encrypted data. Em *Theory of Cryptography Conference (TCC)*, páginas 535–554. Springer.

[Cachin et al. 2009] Cachin, C., Keidar, I., e Shraer, A. (2009). Trusting the cloud. *ACM SIGACT News*, 40(2):81–86.

[Caron et al. 2009] Caron, E., Desprez, F., Loureiro, D., e Muresan, A. (2009). Cloud computing resource management through a grid middleware. Em *IEEE Conference on Cloud Computing*.

[Chow et al. 2009] Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R., e Molina, J. (2009). Controlling data in the cloud: outsourcing computation without outsourcing control. Em *ACM workshop on Cloud Computing Security*, páginas 85–90, New York, NY, USA. ACM.

[CSA 2009] CSA (2009). *Security Guidance for Critical Areas of Focus in Cloud Computing – v2.1*. Cloud Security Alliance.

[CSA 2010a] CSA (2010a). Domain 12: Guidance for identity & access management v2.1. http://www.cloudsecurityalliance.org/guidance/csaguide-dom12-v2.10.pdf.

[CSA 2010b] CSA (2010b). *Top Threats to Cloud Computing V1.0*. Cloud Security Alliance.

[Dalton et al. 2009] Dalton, C. I., Plaquin, D., Weidner, W., Kuhlmann, D., Balacheff, B., e Brown, R. (2009). Trusted virtual platforms: A key enabler for converged client devices. *ACM SIGOPS Operating Systems Review*, 43(1):36–43.

[Dawoud et al. 2010] Dawoud, W., Potsdam, G., Takouna, I., e Meinel, C. (2010). Infrastructure as a service security: Challenges and solutions. Em *7th International Conference on Informatics and Systems (INFOS)*.

[De Capitani di Vimercati e Samarati 2006] De Capitani di Vimercati, S. e Samarati, P. (2006). Privacy in the electronic society. Em *International Conference on Information Systems Security (ICISS)*, Kolkata, India. invited talk.

[Doelitzscher et al. 2010] Doelitzscher, F., Reich, C., e Sulistio, A. (2010). Designing cloud services adhering to government privacy laws. Em *International Symposium on Trust, Security and Privacy for Emerging Applications*.

[Erickson et al. 2009] Erickson, J. S., Spencer, S., Rhodes, M., Banks, D., Rutherford, J., Simpson, E., Belrose, G., e R., R. P. (2009). Content-centered collaboration spaces in the cloud. *IEEE Internet Computing*, páginas 34–42.

[Ernst & Young 2010] Ernst & Young (2010). Insights in it risk – top privacy issues for 2010.

[Etsion et al. 2009] Etsion, Y., Ben-Nun, T., e Feitelson, D. (2009). A global scheduling framework for virtualization environments. Em *IEEE Symposium on Parallel and Distributed Processing*.

[Eucalyptus 2010] Eucalyptus (2010). Eucalyptus – the open source cloud plataform. http://open.eucalyptus.com.

[FIPS 140-2 2001] FIPS 140-2 (2001). *Security Requirements for Criptographic Modules - FIPS PUB 140-2*. Computer Security Division.

[Fischer-Hübner 2001] Fischer-Hübner, S. (2001). IT-Security and Privacy: Design and use of privacy-enhancing security mechanisms. Em Goos, G., Hartmanis, J., e van Leeuwen, J., editores, *Lecture Notes in Computer Science*, volume 1958. Springer-Verlang.

[Foster et al. 2008] Foster, I., Zhao, Y., Raicu, I., e Lu, S. (2008). Cloud computing and grid computing 360-degree compared. Em *Grid Computing Environments Workshop*.

[Gentry 2009] Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. Em *Annual ACM Symposium on Theory of computing*, páginas 169–178, New York, NY, USA. ACM.

[Goyal e Mikkilineni 2009] Goyal, P. e Mikkilineni, R. (2009). Policy-based event-driven services-oriented architecture for cloud services operation & management. Em *IEEE International Conference on Cloud Computing*, páginas 135–138. IEEE Computer Society.

[Grobauer et al. 2010] Grobauer, B., Walloschek, T., e Stöcker, E. (2010). Understanding Cloud-Computing Vulnerabilities. *IEEE Security and Privacy*.

[Grossman 2009] Grossman, R. (2009). The case for cloud computing. *IT Professional*, 11(2).

[Gruschka e Iacono 2009] Gruschka, N. e Iacono, L. (2009). Vulnerable cloud: SOAP message security validation revisited. Em *IEEE International Conference on Web Services*.

[Hayes 2008] Hayes, B. (2008). Cloud computing. *Communications of the ACM*, 51(7):9–11.

[Hinde 2003] Hinde, S. (2003). Privacy legislation: a comparison of the US and european approaches. *Computers & Security*, 22(5):378–387.

[ISO 2005] ISO (2005). *ISO/IEC 27001 - Information technology - Security techniques - Information security management systems - Requirements*. International Organization for Standartization.

[Itani et al. 2009] Itani, W., Kayssi, A., e Chehab, A. (2009). Privacy as a service: Privacy-aware data storage and processing in cloud computing architectures. Em *IEEE International Conference on Dependable, Autonomic and Secure Computing*, páginas 711 –716.

[ITIL 2010] ITIL (2010). *IT Infrastructure Library*. Office of Governance Commerce, UK.

[ITU-T 2000] ITU-T (2000). *Information technology – Open systems interconnection – The Directory: Public-key and attribute certificate frameworks. Recommendation X.509*. International Telecommunication Union.

[Jensen et al. 2009] Jensen, M., Schwenk, J., Gruschka, N., e Iacono, L. (2009). On technical security issues in cloud computing. Em *IEEE International Conference on Cloud Computing*, páginas 109–116. IEEE Computer Society.

[Kaliski Jr e Pauley 2010] Kaliski Jr, B. e Pauley, W. (2010). Toward risk assessment as a service in cloud environments. Em *USENIX Workshop on Hot Topics in Cloud Computing*.

[Kandukuri et al. 2009] Kandukuri, B., Paturi, V., e Rakshit, A. (2009). Cloud security issues. Em *International Conference on Services Computing*.

[Landwehr 2001] Landwehr, C. (2001). Computer security. *International Journal of Information Security*, 1(1):3–13.

[Laureano e Maziero 2008] Laureano, M. e Maziero, C. (2008). Virtualização: Conceitos e aplicações em segurança. Em Maziero, C., editor, *Livro-Texto de Minicursos SBSeg*, páginas 1–50. Sociedade Brasileira de Computação.

[Mather et al. 2009] Mather, T., Kumaraswamy, S., e Latif, S. (2009). *Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance*. O'Reilly Media.

[Mell e Grance 2009] Mell, P. e Grance, T. (2009). The NIST definition of cloud computing. *National Institute of Standards and Technology*.

[Nurmi et al. 2009] Nurmi, D., Wolski, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L., e Zagorodnov, D. (2009). The eucalyptus open-source cloud computing system. Em *IEEE/ACM International Symposium on Cluster Computing and the Grid*.

[OASIS 2005a] OASIS (2005a). *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS.

[OASIS 2005b] OASIS (2005b). *eXtensible Access Control Markup Language version 2.0*.

[OASIS 2005c] OASIS (2005c). *SAML 2.0 profile of XACML v2.0*. OASIS.

[OASIS 2006] OASIS (2006). *Service Provisioning Markup Language (SPML) Version 2*. OASIS.

[OASIS 2009a] OASIS (2009a). *Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of Security Assertion Markup Language (SAML) for Healthcare Version 1.0*. OASIS.

[OASIS 2009b] OASIS (2009b). *Web Services Federation Language version 1.2*. OASIS.

[Ogrizovic et al. 2010] Ogrizovic, D., Svilicic, B., e Tijan, E. (2010). Open source science clouds. Em *International Convention (MIPRO 2010)*.

[OpenID 2010] OpenID (2010). *OpenID Foundation - OIDF*. OpenID Foundation.

[Pearson et al. 2009] Pearson, S., Shen, Y., e Mowbray, M. (2009). A privacy manager for cloud computing. Em Jaatun, M., Zhao, G., e Rong, C., editores, *Cloud Computing*, volume 5931 of *LNCS*, páginas 90–106. Springer. 10.1007/978-3-642-10665-1-9.

[Pedersen 1992] Pedersen, T. (1992). Non-interactive and information-theoretic secure verifiable secret sharing. Em Feigenbaum, J., editor, *Advances in Cryptology*, volume 576 of *Lecture Notes in Computer Science*, páginas 129–140. Springer Berlin / Heidelberg.

[Pfleeger e Pfleeger 2006] Pfleeger, C. P. e Pfleeger, S. L. (2006). *Security in Computing*. Prentice Hall, fourth edition.

[Pinheiro Jr e Kon 2005] Pinheiro Jr, J. e Kon, F. (2005). Segurança em grades computacionais. Em *Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*.

[Provos et al. 2009] Provos, N., Rajab, M., e Mavrommatis, P. (2009). Cybercrime 2.0: when the cloud turns dark. *Communications of the ACM*, 52(4):42–47.

[Rezgui et al. 2003] Rezgui, A., Bouguettaya, A., e Eltoweissy, M. Y. (2003). Privacy on the web: Facts, challenges, and solutions. *IEEE Security and Privacy*, 1(6):40–49.

[Rimal et al. 2009] Rimal, B., Choi, E., e Lumb, I. (2009). A taxonomy and survey of cloud computing systems. Em *International Joint Conference on INC, IMS and IDC*.

[Ristenpart et al. 2009] Ristenpart, T., Tromer, E., Shacham, H., e Savage, S. (2009). Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. Em *ACM conference on Computer and communications security*, páginas 199–212. ACM.

[Shen et al. 2009] Shen, E., Shi, E., e Waters, B. (2009). Predicate privacy in encryption systems. Em Reingold, O., editor, *Theory of Cryptography*, volume 5444 of *Lecture Notes in Computer Science*, páginas 457–473. Springer Berlin / Heidelberg.

[Shirey 2000] Shirey, R. (2000). *RFC 2828 - Internet Security Glossary*. The Internet Society.

[Singh et al. 2010] Singh, M. D., Krishna, P. R., e Saxena, A. (2010). A cryptography based privacy preserving solution to mine cloud data. Em *Annual ACM Bangalore Conference*, páginas 1–4, New York, NY, USA. ACM.

[Somani e Chaudhary 2009] Somani, G. e Chaudhary, S. (2009). Application performance isolation in virtualization. Em *IEEE International Conference on Cloud Computing*, páginas 41–48. IEEE Computer Society.

[Song et al. 2000] Song, D. X., Wagner, D., e Perrig, A. (2000). Practical techniques for searches on encrypted data. Em *IEEE Symposium on Security and Privacy*, páginas 44 –55.

[Stahl 2008] Stahl, B. C. (2008). The impact of the UK Human Rights Act 1998 on privacy protection in the workplace. Em Subramanian, R., editor, *Computer Security, Privacy, and Politics - Current Issues, Challenges, and Solutions*, chapter IV, páginas 55–69. IRM Press.

[Staples 2007] Staples, W. G. (2007). *Encyclopedia of Privacy*. Greenwood Press.

[Sweeney 2002] Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570.

[Trusted Computing Group 2010] Trusted Computing Group (2010). *Trusted Platform Module – Specifications*. Trusted Computing Group.

[Turn e Ware 1975] Turn, R. e Ware, W. H. (1975). Privacy and security in computer systems. Technical Report P5361, Rand Corporation.

[VMWare Inc 2010] VMWare Inc (2010). VMware vSphere. http://www.VMware.com/vSphere.

[W3C 2001] W3C (2001). *XML Key Management Specification (XKMS)*. W3 Consortium.

[W3C 2010] W3C (2010). *Web Services Policy 1.5 – Framework*. W3 Consortium.

[Wang et al. 2010a] Wang, C., Wang, Q., Ren, K., e Lou, W. (2010a). Privacy-preserving public auditing for data storage security in cloud computing. Em *IEEE International Conference on Computer Communications*.

[Wang et al. 2010b] Wang, H., Jing, Q., Chen, R., He, B., Qian, Z., e Zhou, L. (2010b). Distributed systems meet economics: Pricing in the cloud. Em *USENIX Workshop on Hot Topics in Cloud Computing*.

[Wang et al. 2009] Wang, J., Shao, Y., Jiang, S., e Le, J. (2009). Providing privacy preserving in cloud computing. Em *International Conference on Test and Measurement*, páginas 213–216. IEEE Computer Society.

[Ware 1973] Ware, W. H. (1973). Records, computers and the rights of citizens. Technical Report P5077, Rand Corporation.

[Wood et al. 2010] Wood, T., Cecchet, E., Ramakrishnan, K., Shenoy, P., van der Merwe, J., e Venkataramani, A. (2010). Disaster recovery as a cloud service: Economic benefits & deployment challenges. Em *USENIX Workshop on Hot Topics in Cloud Computing*.

[Wright 2004] Wright, T. (2004). *Security, Privacy, and Anonymity*. ACM.

[Yee e Korba 2009] Yee, G. e Korba, L. (2009). Personal privacy policies. Em Vacca, J., editor, *Computer and Information Security Handbook*, páginas 487–505. Morgan Kaufmann.

[Yildiz et al. 2009] Yildiz, M., Abawajy, J., Ercan, T., e Bernoth, A. (2009). A layered security approach for cloud computing infrastructure. Em *International Symposium on Pervasive Systems, Algorithms, and Networks*, páginas 763–767. IEEE.

[Zhang e Zhou 2009] Zhang, L. e Zhou, Q. (2009). CCOA: Cloud computing open architecture. Em *International Conference on Web Services*.

[Zhang et al. 2010] Zhang, Q., Cheng, L., e Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Springer Journal of Internet Services and Applications*, páginas 7–18.

[Zhao et al. 2009] Zhao, Y., Xie, Y., Yu, F., Ke, Q., Yu, Y., Chen, Y., e Gillum, E. (2009). Botgraph: Large scale spamming botnet detection. Em *USENIX Symposium on Networked systems design and implementation*, páginas 321–334. USENIX Association.