UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Ari Silveira Anselmo Junior

GERENCIAMENTO DE IDENTIDADES COMO UM SERVIÇO PARA AMBIENTES DE COMPUTAÇÃO EM NUVEM

Dissertação submetida ao Programa de Pós Graduação em Ciências da Computação da Universidade Federal de Santa Catarina para a obtenção do Grau de Mestre em Ciências da Computação
Orientadora: Profa. Dra. Carla Merkle Westphall

Florianópolis

2011

Catalogação na fonte elaborada pela bibliotecária
Augiza Karla Boso CRB14/1092

A587g Anselmo Junior, Ari Silveira
Gerenciamento de identidades como um serviço para ambientes de computação em nuvem [dissertação] / Ari Silveira Anselmo Junior ; orientadora, Carla Merkle Westphall. – Florianópolis, SC, 2011.
1 v. : il., graf. , tabs.

Dissertação (mestrado) – Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação.

Inclui referências.

1. Informática. 2. Ciência da computação. 3. Gerenciamento de identidades. 4. Computação em nuvem. I. Westphall, Carla Merkle. II. Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU: 681

Ari Silveira Anselmo Junior

GERENCIAMENTO DE IDENTIDADES COMO UM SERVIÇO PARA AMBIENTES DE COMPUTAÇÃO EM NUVEM

Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Ciência da Computação – Área de Concentração Ciência da Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação

Florianópolis, 28 de fevereiro de 2011.

_____
Prof. Dr. Mário Antônio Ribeiro Dantas
Coordenador do Curso

Banca Examinadora:

_____
Carla Merkle Westphall, Dra.
Orientadora
Universidade Federal de Santa Catarina

_____
Patricia Della Méa Plentz, Dra.
Universidade Federal de Santa Catarina

_____
Rômulo Silva de Oliveira, Dr.
Universidade Federal de Santa Catarina

_____
Alexandre Moraes Ramos, Dr.
Universidade Federal de Santa Catarina

A minha noiva Augiza que sempre esteve ao meu lado durante todo o caminho.
A minha mãe Lucia pelo amor, incentivo e apoio constantes.

“Jamais considere seus estudos como uma obrigação, mas como uma oportunidade invejável para aprender a conhecer a influência libertadora da beleza do reino do espírito, para seu próprio prazer pessoal e para proveito da comunidade à qual seu futuro trabalho pertencer**”.**

(Albert Eistein)

AGRADECIMENTOS

Meus profundos e sinceros agradecimentos a professora Carla Merkle Westphall, pela forma paciente com que me orientou sempre me apoiando, disposta a ajudar nos mais variados dias e horários e sem a qual eu não conseguiria alcançar meus objetivos.

Agradeço aos membros da banca pelas ótimas dicas e correções que foram essenciais para a qualidade final deste trabalho.

Agradeço aos meus colegas da Dataprev pelos conhecimentos e conselhos e por sempre ouvirem minhas descobertas e frustrações ao longo desse período.

Agradeço aos meus pais e aos meus irmãos, terem me incentivado a sempre seguir em frente.

A minha noiva Augiza que sempre esteve ao meu lado e pela compreensão depositada nessa jornada de noites em claro e finais de semana dedicados a conclusão deste trabalho.

RESUMO

O gerenciamento de identidades digitais esteve presente nos últimos anos como um grande problema para os usuários e um desafio intrínseco para pesquisadores. A dificuldade em gerenciar um grande número de identidades, a baixa usabilidade e o alto custo de armazenamento e manutenção dos sistemas de autenticação levou a análise e construção das mais variadas soluções para o gerenciamento do ciclo de vida das identidades digitais.

Computação em nuvem surgiu como um novo paradigma que visa prover acesso a recursos computacionais de maneira dinâmica e escalável. Como conseqüência, novos desafios de segurança emergiram, deixando evidente a necessidade de novas soluções. Em um ambiente onde potencialmente várias organizações compartilham os mesmos recursos computacionais, é de vital importância estabelecer fronteiras e limites para o acesso a dados e serviços, mas a preocupação com a privacidade das informações de identificação compartilhadas na nuvem vem se mostrando um grande entrave na ampla adoção do ambiente.

Com as organizações migrando para arquiteturas orientadas a serviço na nuvem, os componentes responsáveis pelo gerenciamento de identidade devem ser separados das aplicações, disponíveis por padrão, e oferecidos também como um serviço nesta arquitetura. Desta maneira é possível fácil reuso, economia de recursos e diminuição de pontos de falha.

Este trabalho propõe uma arquitetura de gerenciamento de identidades como serviço para o ambiente de computação em nuvem, centrado no usuário de forma a tratar os problemas intrínsecos deste ambiente. Para isto é utilizado um modelo centralizado entre provedores de serviço hospedados na mesma nuvem e uma federação de identidades entre nuvens, com controle do usuário sobre as informações de identificação pessoais compartilhadas com os provedores de serviço. A arquitetura proposta é apoiada em padrões existentes e amplamente adotados para o gerenciamento de identidades.

Palavras-chave: Gerenciamento de Identidade, Computação em nuvem, Federação de Identidades, Gerenciamento de Identidade como um serviço.

ABSTRACT

The management of digital identities has been present in recent years as a major problem for users and an intrinsic challenge for researchers. The difficulty in managing a large number of identities, low usability and high cost of storage and maintenance of authentication systems has led to the analysis and construction of several solutions for managing the lifecycle of digital identities.

Cloud computing has emerged as a new paradigm that aims to provide access to computational resources in a dynamic and scalable way. As a result, new security challenges have emerged, leaving a clear need for new solutions. In an environment where potentially multiple organizations share the same computing resources, it is vitally to set boundaries and limits on data and services access, but concern for privacy of identifiable information shared in the cloud is proving a major obstacle in the wide adoption of the cloud environment.

With organizations moving to service oriented architectures in the cloud, components responsible for identity management should be separated from the applications, available by default, and also offered as a service in this architecture. Providing easily reuse, saving resources and reducing points of failure.

This paper proposes a architecture of identity management as a service for cloud computing environment in a user-centered way for addressing the problems inherent in this environment. In order to achieve this goal is used a centralized model between service providers in the same cloud and a identity federation between clouds. The model is supported by existing and widely adopted standards for identity management.

Keywords: Identity Management, Cloud Computing, Identity Federation, Identity Management as a service.

LISTA DE FIGURAS

Figura 1: Modelo Centralizado 24
Figura 2: Asserção SAML 28
Figura 3: Definições Computação em Nuvem 39
Figura 4: Arquitetura Eucalyptus 42
Figura 5: Sistema descentralizado na nuvem 47
Figura 6: IdaaS como modelo de IDM centralizado em um mesmo CP 50
Figura 7: Circulo de Confiança entre nuvens 52
Figura 8: Requisição acesso as PIIs 54
Figura 9: Tela inicial processo de instalação UEC 57
Figura 10: Seleção Modo de Instalação 58
Figura 11: Processador sem suporte a virtualização por hardware 59
Figura 12: Tela de acesso UEC 60
Figura 13: Tela de seleção e instalação de imagens 60
Figura 14: Configurando a url da nuvem privada no HybridFox 61
Figura 15: Credenciais 62
Figura 16: Inserindo Credenciais 62
Figura 17: HybridFox Configurado 63
Figura 18: Estrutura Nuvem 1 65
Figura 19: Extrutura Nuvem 2 67
Figura 20: Tela de registro de SPs OpenAM 68
Figura 21: Tela Cadastro PIIs IdaaS 70
Figura 22: Funcionamento do processo de SSO utilizando o IDaaS 71
Figura 23: Tela autenticação IdaaS 72
Figura 24: Liberação das PIIs 72
Figura 25: Aplicação 2 com acesso as PIIs 73

LISTA DE TABELAS

Tabela 1: Hardware Nuvem 1 64
Tabela 2: Hardware Nuvem 2 66
Tabela 3: PIIs Compartilhadas e Níveis de Acesso 69

LISTA DE ABREVIATURAS E SIGLAS

CAFe – Comunidade Acadêmica Federada
CC – Cluster Controller
CLC – *Cloud Controller*
CP – Provedor de Nuvem (*Cloud Provider*)
IaaS – Infraestrutura como serviço (*Infrastructure as a Service*)
IdaaS – Gerenciamento de Identidade como serviço (*Identity As A Service*)
IdM – Gerenciamento de Identidade
IdP – Provedor de Identidade (*Identity provider*)
LDAP – *Lightweight Directory Access Protocol*
PA – *Policy Agents*
PaaS – Plataforma como serviço (*Plataform as a Service*)
SaaS – Software como serviço (*Software as a Service*)
SC – Storage Controller
SLA – Acordo de Nível de Serviço (Service Level Agreement)
SOA – Arquitetura Orientada a Serviço (*Service-oriented architecture*)
SOAP - *Simple Object Access Protocol*
SP – Provedor de serviço (*Service Provider*)
SSO – Login Único (*Single Sign On*)
TI – Tecnologia da Informação
UDDI – *Universal Description, Discovery and Integration*
UEC – Ubuntu Enterprise Cloud
UEC – Ubuntu Enterprise Cloud
URI – *Uniform Resource Identifier*
UUID – *Universal Unique IDentifier*
XACML – *eXtensible Access Control Markup Language*
XML – *eXtensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO 17

1.1 OBJETIVO GERAL 18
1.2 OBJETIVOS ESPECÍFICOS 18
1.3 TRABALHOS RELACIONADOS 19
1.4 ORGANIZAÇÃO DO TRABALHO 20

2 GERENCIAMENTO DE IDENTIDADES 21

2.1 CONCEITOS BÁSICOS DE GERENCIAMENTO DE IDENTIDADE E CLASSIFICAÇÃO 22
2.2 IDENTIDADES FEDERADAS 25
2.2.1 Protocolos de Federação 27
2.2.2 Vantagens na Utilização de Identidades Federadas 30
2.3 SOLUÇÕES DE GERENCIAMENTO DE IDENTIDADES 31
2.3.1 Shibboleth 31
2.3.2 OpenAM 31

3 COMPUTAÇÃO EM NUVEM 34

3.1 CONTEXTO E DEFINIÇÃO 34
3.2 ANALOGIA COM O MUNDO REAL 34
3.3 CARACTERÍSTICAS DA NUVEM 36
3.5 MODELOS DE SERVIÇO NA NUVEM 37
3.6 MODELOS DE NUVEM 38
3.7 FERRAMENTAS RELEVANTES 40
3.7.1 Eucalyptus 40
3.8 QUESTÕES DE SEGURANÇA NA NUVEM 42

4. GERENCIAMENTO DE IDENTIDADE COMO SERVIÇO 44

(IDAAS - *IDENTITY AS A SERVICE*) 44

4.1 IDENTIDADE NA NUVEM 44
4.2 PROPOSTA: GERENCIAMENTO DE IDENTIDADE NA NUVEM 48
4.2.1 Gerenciamento de Identidades Externalizado como Serviço 48
4.2.2 Provedor de Nuvem como terceira Parte Confiável 50
4.2.3 Provisionamento dos usuários pelo Provedor de Nuvem 51
4.2.4 SSO 51
4.2.5 Federação de identidades entre Provedores de Nuvem 51

4.2.6 Controle das identidades centrado ao usuário e controle das PIIs compartilhadas com os provedores de serviços *53*
4.2.7 Níveis de liberação das Informações Pessoais de Identificação *54*

5 VALIDAÇÃO **56**

5.1 FERRAMENTAS UTILIZADAS 56
5.2 IMPLANTANDO O AMBIENTE DE TESTES 56
5.2.1 Processo de Instalação Ubuntu Enterprise Cloud *57*
5.2.2 Gerenciando as Instâncias na Nuvem *61*
5.3 IMPLANTAÇÃO E FUNCIONAMENTO 63
5.3.1 Registro dos SPs no IDaaS *67*
5.3.2 Cadastro dos usuários *68*
5.3.3 Cadastro e manutenção dos usuários e PIIs *68*
5.3.4 Funcionamento do processo de autenticação *70*
5.3.5 Liberação das PIIs *72*

6. CONCLUSÃO E TRABALHOS FUTUROS **74**

REFERÊNCIAS **76**

1 INTRODUÇÃO

O gerenciamento de identidades digitais esteve presente nos últimos anos como um grande problema para os usuários e um desafio intrínseco para pesquisadores. A dificuldade em gerenciar um grande número de identidades, a baixa usabilidade e o alto custo de armazenamento e manutenção dos sistemas de autenticação e autorização levou a análise e construção das mais variadas soluções para o gerenciamento do ciclo de vida das identidades digitais.

A computação em nuvem surgiu como um novo paradigma que visa prover acesso a recursos computacionais de maneira dinâmica e escalável [Leaf 2010] [Buyya et al. 2008]. Como consequência, novos desafios de segurança emergiram, deixando evidente a necessidade de novas soluções.

O modelo de serviço, modo de operação e as tecnologias utilizadas para prover os serviços em um ambiente de computação em nuvem apresentam diferentes níveis de risco quando comparado ao ambiente tradicional de tecnologia da informação [CSA 2009].

Em um ambiente onde potencialmente várias organizações compartilham os mesmo recursos computacionais, é de vital importância estabelecer fronteiras e limites para o acesso a dados e serviços, mas a preocupação com a privacidade das informações de identificação compartilhadas na nuvem vem se mostrando um grande entrave na ampla adoção do ambiente [CSA 2010b].

Com as organizações migrando para arquiteturas orientadas a serviço na nuvem [Zhang, Zhou 2009] os componentes responsáveis pelo gerenciamento de identidade podem ser componentes separados das aplicações, e oferecidos como um serviço também nesta arquitetura [CSA 2010] [Mather et al. 2009]. Desta maneira possibilita fácil reuso, economia de recursos e diminui pontos de falha.

Esse trabalho propõe um modelo de gerenciamento de identidades como serviço (*Identity as a Service* - IdaaS) para o ambiente de computação em nuvem no modelo de Software como Serviço, centrado no usuário e adaptado para tratar os problemas intrínsecos deste ambiente. Oferece serviços já comuns no modelo de computação tradicional, como autenticação única (*Single Sign-On* - SSO) e federação de identidades para aplicações hospedadas na nuvem. Para isto é utilizado um modelo de identidade centralizado no contexto interno de uma nuvem como também federação de identidades com

formação de círculos de confiança entre provedores de nuvem distintos. Para o controle de privacidade dos dados de identificação é proposto o controle do usuário sobre essas informações compartilhadas com os provedores de serviço na nuvem. O modelo é apoiado em padrões existentes e amplamente adotados pela indústria para o gerenciamento de identidades.

1.1 Objetivo Geral

O objetivo geral deste trabalho é propor um modelo de gerenciamento de identidades como um serviço para ambiente de computação em nuvem, pensando nos problemas específicos inerentes ao domínio que considere a proteção dos dados de autenticação e identificação, visando à proteção da privacidade dos usuários utilizadores de serviços na nuvem.

1.2 Objetivos Específicos

Os objetivos específicos que podem ser citados são:

a) Estudo e identificação dos principais problemas e soluções no gerenciamento de identidades;
b) Investigação dos problemas do gerenciamento de identidades no ambiente de computação em nuvem;
c) Analise das técnicas e ferramentas existentes para o gerenciamento de identidade e proposição de soluções para o desenvolvimento e uso de gerenciamento de identidades para o ambiente de computação em nuvem;
d) Proposição de mecanismos que considerem questões de privacidade dos usuários e proteção das informações de identificação pessoal (PII – *Personally Identifiable Information*);
e) União das soluções propostas para o gerenciamento de identidade para computação em nuvem;
f) Montar e configurar um ambiente de Computação em Nuvem no qual serão realizados testes da proposta;
g) Construção de um protótipo para demonstrar e avaliar as idéias investigadas, concebidas e propostas.

1.3 Trabalhos Relacionados

O interesse da comunidade científica pelo tema de computação em nuvem é relativamente recente, mas a preocupação com o gerenciamento de identidades e controles de privacidade dos dados de autenticação e autorização na nuvem já chamou a atenção dos pesquisadores e vem originando diversos trabalhos e pesquisas na área.

Em [Celesti et al. 2010] é proposto um modelo de autenticação entre provedores de nuvem, utilizando uma terceira parte confiável e buscando dessa forma a troca de informações de autenticação entre os provedores de serviço hospedados na mesma.

Semelhantemente, [Huang et al. 2010] propõe um mediador (*broker*) de identidades federadas para nuvem que utiliza uma terceira parte confiável para facilitar o controle da federação.

Em [Jaeger et al. 2010] sugere-se um gerenciador de privacidade utilizando técnicas de ofuscação para proteger os dados do usuário contra uso indevido pelo provedor de nuvem, dessa maneira os dados nunca estariam acessíveis aos provedores que os armazenam sem que o usuário se autentique e forneça a chave necessária para utilização dos dados.

O trabalho [Pearson et al. 2009] propõe um modelo de privacidade como um serviço (PaaS), para assegurar a privacidade dos dados hospedados na nuvem utilizando processadores criptográficos.

Além dos trabalhos já publicados no meio científico, pesquisadores do meio privado demonstram que suas equipes estão em busca de soluções para os problemas na nuvem. Uma possibilidade interessante é a definição da implementação do serviço de gerenciamento identidades como um componente externo às aplicações.

André Durand, fundador e CEO da Ping Identy em seu Keynote Identity in the Cloud na European Identity Conference 2010 descreve a importância de externalizar os serviços de federação de identidades na nuvem e de se usar padrões existentes para isso, convocando pesquisadores em todo mundo a seguir o mesmo caminho.

Da mesma forma, Nishant Kaushik, arquiteto de produto e estrategista líder para soluções em gerenciamento de identidade Oracle, em sua apresentação Identity Management and the Cloud: Stormy Days Ahead na Oracle OpenWorld 2010 mostrou que sua equipe pesquisa atualmente formas de prover identidade na nuvem de uma maneira reusável e orientada a serviço.

Marco Casassa Mont cientista pesquisador do Hewlett Packard Lab em sua apresentação na EEMA e-Identity Conference disserta sobra a importância vital do gerenciamento de identidades na nuvem e sobre riscos da falta de padronização principalmente entre provedores neste ambiente.

1.4 Organização do Trabalho

O capítulo 2 inicia com uma revisão bibliográfica sobre gerenciamento e federação de identidades. No capitulo 3 é feita uma revisão sobre Computação em Nuvem, seu contexto, definição e as principais características de um ambiente em nuvem. A segunda metade do capitulo discorre sobre os principais problemas de segurança na nuvem e os riscos para a privacidade dos usuários. No capítulo 4 são descritos os desafios do gerenciamento de identidades na nuvem, são apresentadas as características necessárias do modelo e também é definido o modelo proposto. No capítulo 5 é mostrado como foi montado o ambiente de testes e o funcionamento do protótipo criado. No capítulo 6 são descritas a conclusão e trabalhos futuros.

2 GERENCIAMENTO DE IDENTIDADES

Diariamente temos a necessidade de nos identificar em diversos domínios, sendo comum que um usuário tenha que criar e manter vários pares usuário-senha para executar tarefas corriqueiras. Essa necessidade rotineira de autenticação, além de ser um incômodo para o usuário, provoca um grande número de entraves aos prestadores de serviços, entre eles, o custo de manter e armazenar dados de diversos sistemas de autenticação.

O crescente número de sistemas conectados se mostrou um desafio à usabilidade, principalmente a de aplicativos web. Mesmo para um usuário com experiência em lidar com identidades e métodos de autenticação, o gerenciamento das identidades digitais nos diversos domínios pode ser muito mais complexo do que identidades do mundo real. Identidades digitais não estão vinculadas para sempre ao seu proprietário, Luiz Vicente sempre será Luiz Vicente, mas vicente@nspd.com pode perder a validade ou até mesmo vir a identificar outro usuário.

Grandes organizações estão em uma busca constante para dar maior usabilidade aos seus aplicativos, com o objetivo de aumentar assim seu público alvo. Um sistema de autenticação incômodo, repetitivo e pouco transparente forma uma barreira aos usuários do sistema, criando um difícil paradoxo, pois dada a importância vital da fase de autenticação, a mesma não pode ser ignorada. As conseqüências da divulgação de informações sigilosas ou do acesso ilegal a serviços podem ser catastróficas para qualquer empresa, independente do porte. Neste contexto surgiram os sistemas de gerenciamento de identidades.

Definição

O gerenciamento de identidades consiste em um conjunto de funções e habilidades, como administração, descoberta e troca de informações, usadas para garantir a identidade de uma entidade e as informações contidas nessa identidade, permitindo assim que relações comerciais possam ocorrer de forma segura [Chadwick 2009]. Uma boa solução de gerenciamento de identidade deve ser capaz de elevar a produtividade, permitir maior eficiência na entrega dos serviços computacionais, diminuir custos e elevar a fidelidade dos utilizadores.

2.1 Conceitos básicos de gerenciamento de identidade e Classificação

Para podermos classificar corretamente os sistemas de gerenciamento de identidades precisamos primeiro definir alguns termos comumente utilizados neste contexto.

i. O **usuário** é a pessoa que assume a identidade digital que será usada para interagir com uma aplicação digital.

ii. **Identidade Digital.** Uma identidade digital é formada por um identificador único e atributos descritivos. Por exemplo, a conta de um usuário no sistema operacional.

iii. **Credenciais.** Credenciais são informações relacionadas ou derivadas da posse de uma identidade digital, geralmente secretas e apenas o portador da identidade deve conhecê-la.

iv. **Autenticação.** Processo que verifica as credenciais de uma identidade digital contra os valores de uma base de identidades.

v. **Autorização.** É o processo de decisão que verifica os direitos do usuário com as permissões configuradas para o acesso a um recurso, permitindo o controle de acesso.

vi. **Domínio de segurança**. Limites lógicos ou físicos de uma organização que definem uma área de acesso confiável e administração de segurança.

vii. O **Provedor de Serviço** (SP, do inglês *service provider*) é uma aplicação web que possui o serviço ou informação que o usuário deseja obter acesso. O provedor de serviço delega o serviço de autenticação a uma terceira parte, o provedor de identidade, que pode enviar ao SP informações sobre o usuário.

viii. O **Provedor de Identidade** (IdP, do inglês *identity provider*) é responsável por emitir a identidade de um usuário. Após o usuário passar por um processo de autenticação, este recebe uma credencial, que é reconhecida como válida pelos provedores de serviço.

Os sistemas de gerenciamento de identidades podem ser clasificados como tradicional, centralizado e federado [Bhargav-Spantzel et al. 2007].

Modelo Tradicional

No modelo tradicional, a identificação do usuário é tratada de forma isolada por cada provedor de serviços, o qual também atua como provedor de identidades. É o modelo mais comum, ficando a cargo do usuário criar uma identidade digital para cada sistema com o qual tenha interesse em acessar. Este modelo cria grandes custos para os provedores de serviço e um enorme peso para o usuário que deve manter várias identificações, como por exemplo, pares usuário-senha. Esta tarefa repetitiva de sempre fornecer as mesmas informações no momento da criação de sua identidade, faz com que o usuário não seja tão fiel no preenchimento de atributos que não são cruciais para acessar o recurso oferecido pelo provedor de serviços [Fraga et al. 2010].

Modelo Centralizado

O modelo centralizado surgiu como uma solução para a inflexibilidade do modelo tradicional e está fundamentado no compartilhamento das identidades dos usuários entre provedores de serviços e no conceito de autenticação única ou *Single Sign-On* (SSO) [Bhargav-Spantzel et al. 2007]. Como mostrado na figura 1, uma vez autenticado no provedor de identidade (IDP), o usuário tem acesso aos provedores de serviço (SP) no dominio sem ter que repetir o proceso de autenticação para cada um deles. Difere sobre o modelo federado por ter um dominio único de segurança provido por um único IDP.

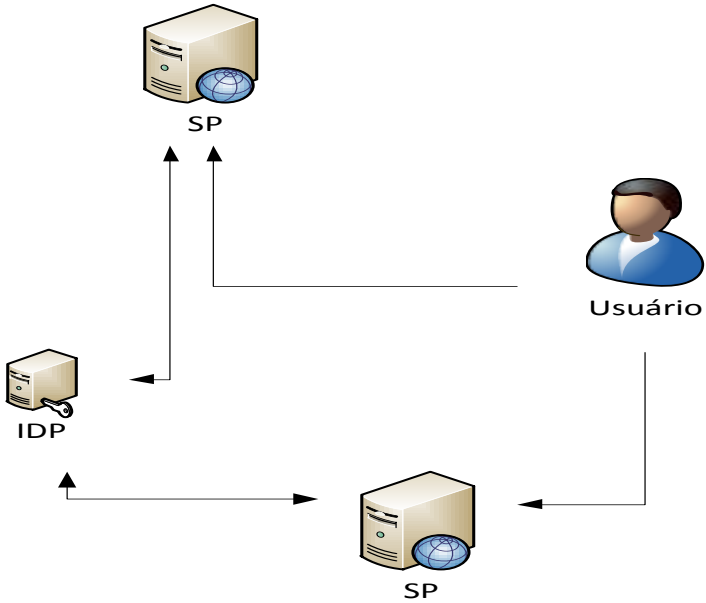

Figura 1: Modelo Centralizado

Modelo Federado

O modelo de identidade federada está fundamentado sobre a distribuição da tarefa de autenticação dos usuários por múltiplos provedores de identidades, estando estes dispostos em diferentes domínios administrativos. Este modelo ganhou grande força nos últimos anos e por suas características vem se mostrando um modelo bastante adequado para resolver alguns problemas do ambiente de computação em nuvem [CSA 2010] e por esse motivo precisamos analisar as características deste modelo mais a fundo.

2.2 Identidades Federadas

O gerenciamento de identidades federadas é um conjunto de tecnologias e processos que permitem que sistemas computacionais distribuam informação de identidade através de diferentes domínios [Eve, Drummond 2008].

Em ciência política, federação refere-se a "um estado que compreende várias regiões parcialmente independentes, unidas por um governo central". Na computação federações de identidades são conceitualmente similares às suas contrapartes não técnicas, com as seguintes diferenças [Armbrust et al. 2010]:

i. As "regiões" ou domínios de uma federação de identidades são normalmente definidos pelos limites físicos, de rede e lógicos de uma única infraestrutura de segurança da organização.

ii. Dentro de uma federação de identidades, as infraestruturas de segurança das várias organizações são quase sempre autônomas e não necessariamente unidas por uma organização central.

A federação funciona basicamente como a associação de parceiros que usam um conjunto comum de atributos, práticas e políticas para troca informações e compartilhamento de serviços, possibilitando a cooperação e transações entre os membros da federação [Carmody et al. 2005].

Uma interessante maneira de ilustrar os princípios da federação de identidades é usar uma analogia com a utilização do passaporte em viagens internacionais [Armbrust et al. 2010]. Se você quiser viajar ao exterior, precisará de um passaporte, caso contrário à segurança de fronteira do país estrangeiro não permitirá sua entrada. A identificação local recebida no seu país natal, como a carteira de identidade, não fornecerá prova que confirme sua identidade em outro país ou talvez não possa ser usada devido às diferenças de idioma. Isso cria a necessidade de um formato comum de credenciais, aceito em vários países, representado pelo passaporte. Para obter um passaporte que permita a viagem para outros países, é preciso primeiramente comprovar sua identidade com outra forma de identificação, como por exemplo, uma certidão de nascimento. Após ter sua identidade comprovada e de seu país local ter definido que não há nenhum tipo de

restrição que impeça sua saída do país, por qualquer motivo, o passaporte será emitido.

O mesmo conceito pode ser aplicado à federação de identidades digitais. O aplicativo cliente acessa um recurso localizado em outro domínio para executar uma transação ou trocar informações. Para tanto, o aplicativo cliente deve ter autorização de seu próprio domínio de segurança para evitar a ocorrência de transações indesejadas e obter credenciais utilizáveis no domínio de segurança onde está localizado o recurso. O domínio não apenas controla o acesso pela validação de credenciais emitidas em outro domínio de segurança, mas também aplica o controle sobre quais recursos os clientes podem acessar em um determinado domínio.

Para descrever o processo de federação precisamos adicionar mais alguns termos a nossa lista de definições:

i. **Token de segurança** – Conjunto de declarações sobre uma parte envolvida em uma transação que pode ser usada para várias finalidades. De modo típico, tokens de segurança são usados para identificar e autenticar a parte representada no token. Tipos diferentes de tokens de segurança incluem diferentes tipos de declarações. Muitos tipos de token de segurança são definidos por um conjunto de padrões estabelecidos para fornecer um formato de token interoperável [Marcon et al. 2010].
ii. **Asserção** – Asserções são instruções sobre uma parte autenticada incluída em um token de segurança emitido. Os tipos de declarações incluídos em um token de segurança dependem, quase sempre, do tipo e da finalidade do token de segurança. As declarações são usadas para várias finalidades, de acordo com o seu tipo. Por exemplo, uma chave criptográfica pode ser incluída como uma declaração para comprovar a posse de uma senha compartilhada ou informações de identificação podem ser incluídas para tomar decisões de autorização sobre a parte representada no token de segurança com base naqueles papéis dos quais fazem parte.
iii. **PII** (*Personally Identifiable Information*) – Informações pessoais que podem ser usadas para identificar um usuário.

No contexto das federações é necessário estabelecer uma relação de confiança (círculo de confiança) entre provedores de identidades e de serviços. Garantindo desta maneira que os SPs terão certeza que um usuário foi autenticado corretamente junto ao provedor de identidades e que o mesmo fornecerá os atributos relacionados ao usuário de forma correta. As asserções emitidas pelo provedor de identidades são assinadas pelo mesmo, o que permite que o provedor de serviços verifique a autenticidade das mesmas [Chadwick 2009].

2.2.1 Protocolos de Federação

Existem vários protocolos que podem ser utilizados na implementação da federação de identidades. Algumas das principais são o SAML, o OpenID, o recurso *cross-forest* do *Active Directory* e ADFS. Daremos uma maior ênfase ao SAML, pois é uma das especificações mais usadas, se tornando um padrão na industria.

Security Assertion Markup Language (SAML)

Lançado pela *OASIS International Consortium*, uma organização que tem como objetivo a criação de padrões baseados em XML. SAML é uma especificação que define uma forma padrão para representar autenticação, atributos e informações de autorização que podem ser utilizados em ambientes distribuídos por aplicativos diferentes [OASIS 2007].

O coração da especificação SAML é o esquema XML que define a representação dos dados de segurança que podem ser usados como parte de uma solução geral para passar o contexto de segurança entre as aplicações.

A SAML define um esquema padrão para tokens de segurança e as definições de mensagens usadas para solicitação, emissão, extensão, término e confirmação na troca de dados de autenticação e autorização entre domínios por forma de asserções, oferecendo assim um modelo flexível e extensível para a propagação de declarações de segurança [OASIS 2005a]. Como já mencionado, a SAML é utilizada tanto em produtos comerciais como acadêmicos, como padrão de definição das mensagens de token.

As **asserções** (*assertion*) SAML [OASIS 2005a] possibilitam a propagação de declarações de segurança podendo ser utilizadas para

resolver o problema de SSO entre múltiplos domínios. Elas são definidas por uma gramática XML que especifica o formato para representar as informações de segurança do cliente em questão. Essas informações por sua vez são testadas por uma parte declarante (*asserting party*) ou autoridade SAML (SAML authority) [Marcon et al. 2010]. Figura 2 apresenta um exemplo de asserção SAML.

```
<saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    Version="2.0" IssueInstant="2011-01-01T12:00:00Z">
    <saml:Issuer Format="urn:oasis:names:SAML:2.0:nameid-format:entity">
        http://teste.lrg.ufsc.br
 </saml:Issuer>
    <saml:Subject>
        <saml:NameID Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress">
            teste@lrg.ufsc.br
 </saml:NameID>
    </saml:Subject>
    <saml:Conditions NotBefore="2011-01-01T13:00:00Z"
        NotOnOrAfter="2011-01-01T13:15:00Z" />

    <saml:AuthnStatement AuthnInstant="2011-01-01T13:00:00Z"
        SessionIndex="65765778792">
        <saml:AuthnContext>
            <saml:AuthnContextClassRef>
                urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
            </saml:AuthnContextClassRef>
        </saml:AuthnContext>
    </saml:AuthnStatement>
</saml:Assertion>
```

Figura 2: Asserção SAML

No exemplo acima podemos identificar os seguintes elementos SAML:

i. ***Issuer***: Possui informações sobre a entidade que está fazendo o pedido de asserção.
ii. ***Subject***: Elemento opcional que especifica o assunto das declarações na asserção.
iii. ***Conditions***: Elemento opcional. Deve ser avaliada quando se usa uma asserção.
iv. ***Authentication statement***: criada pela entidade que realizou com sucesso a autenticação do usuário; a declaração indica quando a autenticação foi feita e qual o mecanismo utilizado (usuário-senha, assinatura digital, etc.);

Além de uma declaração de autenticação (*Authentication*

statement), asserções podem conter diversas declarações da mesma autoridade sobre o mesmo sujeito. Os seguintes tipos de declaração também estão disponíveis:

i. ***Attribute statement***: contém detalhes específicos sobre o sujeito, como nome e certificado digital.

ii. ***Authorization decision statement***: indica ações que o sujeito possui o direito de executar sobre determinados recursos.

A partir da versão 2.0 o SAML também provê suporte ao uso de pseudônimos, que são identificadores dinâmicos e não relacionados a atributos de identidade do sujeito. Por trás do uso dos pseudônimos estão dois protocolos SAML: o protocolo de gerenciamento de identificador de nome e o protocolo de mapeamento de identificador de nome, bem como os protocolos de transporte e perfis associados [OASIS 2005c]. Os pseudônimos servem como identificadores compartilhados entre SP e IdP e podem ser usados de forma persistente, que é quando ele é associado permanentemente ao sujeito, ou de forma transiente, sendo este essencial para o uso em sistemas onde a privacidade é um fator crucial.

O pseudônimo transiente é criado pelo IdP e associado à identidade do usuário pelo tempo que durar o contexto de segurança, permitindo ao SP decidir se o acesso é válido com base em atributos emitidos pelo IdP sendo vedado manter informações sobre o usuário que persistam por mais de uma sessão. SPs diferentes não podem correlacionar acessos do mesmo sujeito, garantindo certo nível de anonimato [OASIS 2005c].

OpenID

OpenID é um sistema de identificação que permite aos usuários se autenticarem em vários sites diferentes usando uma única identidade digital baseada em endereço eletrônico, eliminando a necessidade de manter vários usuários e senhas para cada domínio, é possível até mesmo utilizar uma identidade já existente. É um padrão descentralizado e aberto desenvolvido por Brad Fitzpatrick do *LiveJournal* e mantido pela comunidade de desenvolvedores. OpenID

suporta tanto URL quanto XRIs (*Extensible Resource Identifier*) [OASIS 2005] como identificadores de usuários podendo ainda serem públicos ou privados. Somente a partir da versão 2.0 possui suporte a federação de identidades.

A solução OpenID não possui uma autoridade central que aprova ou registra as partes confiantes (*relying parties*), sítios web, ou provedores OpenID (*OpenID providers*), ficando a cargo do usuário escolher qual provedor OpenID usará (Google, Yahoo, etc) e pode preservar seu identificador caso deseje futuramente mudar de provedor OpenID.

OpenID ganhou relativa aprovação na web, e possui hoje mais de um bilião de contas de usuários habilitados e mais de 50.000 sites que aceitam logins OpenID. Várias organizações de grande porte emitem ou aceitam OpenIDs, incluindo o Google, Facebook, Yahoo, Microsoft, AOL, MySpace, Sears, Universal Music Group, a France Telecom, Novell, Sun e Telecom Italia [OpenID 2010].

Active Directory Federation Service (ADFS)

O ADFS é um componente do Windows Server que fornece SSO interoperável e propagação de declarações de identidade no âmbito da empresa e através dos domínios organizacionais. O ADFS oferece uma solução de federação de segurança interoperável e é parte integrante do Windows Server 2003 R2 e superiores.

2.2.2 Vantagens na Utilização de Identidades Federadas

O uso de federação de identidades permite uma melhor interação com o usuário ao permitir que o processo de autenticação seja menos freqüente e de maneira mais robusta, uma vez que o sistema de autorização pode funcionar de maneira centralizada e focada no processo.

Organizações podem compartilhar informações para atingir suas metas de negócio e oferecer um produto final de maior qualidade, aumentando assim sua vantagem competitiva, além de poder dar um enfoque maior ao serviço em si.

Organizações parceiras podem simplificar o acesso aos seus serviços de maneira mútua e de forma transparente e agradável para o usuário, aumentando assim a confiança e lealdade dos clientes aos

serviços oferecidos. Em um mundo cada vez mais conectado, onde se torna difícil distinguir fronteiras entre empresas e domínios, a federação de identidades tem obtido notável destaque.

2.3 Soluções de Gerenciamento de Identidades

Existem hoje diversas soluções de gerenciamento de identidades federadas para o ambiente tradicional de computação, como exemplo veremos o Shibboleth e OpenAM. Estas soluções utilizam, entre outras, a especificação SAML para a troca dinâmica de informações de segurança.

2.3.1 Shibboleth

O Shibboleth é um projeto da *Internet 2 Middleware Inititative* com uma arquitetura e implementacão baseada na especificacão SAML. O projeto foi proposto com o intuito de resolver os problemas de compartilhamento de recursos de uma forma segura. As suas entidades interagem entre si, de forma a proporcionar um ambiente confiável entre os participantes. O Shibboleth é a solucao de autorização mais empregada para constituição de federações acadêmicas [Moreira et al. 2010]. As federações Incommon, da rede norte-americana Internet 2, e a CAFe, da Rede Nacional de Pesquisa (RNP) do Brasil, são exemplos de federações construídas tendo como o Shibboleth como base. Estas federações agrupam pessoas do meio acadêmico como alunos, técnicos administrativos e professores. Tal comunidade é o principal público alvo de muitas empresas, o que torna interessante a essas empresas o ingresso nas federações acadêmicas também como provedores de serviços.

2.3.2 OpenAM

OpenAM fornece serviços essenciais de gerenciamento de identidade para simplificar a implementação de SSO de forma transparente. Funciona como um componente de segurança gratuito e de código aberto em uma infraestrutura de rede [OpenAM 2010]. Totalmente escrito em java, o OpenAM é baseado no OpenSSO, projeto que era mantido pela SUN e que foi paralizado após a aquisição da empresa pela Oracle. Agora, mantido pela empresa ForgeRock, é um

exemplo de projeto OpenSource com ciclo de vida que se estende além da duração da empresa que o originou.

Tem como objetivo a abstração de serviços de segurança e a manipulação, validação e emissão de tokens de segurança. Tem suporte a SAML 2.0 e permite a formação de círculos de confiança entre SPs. Suporta ainda o provisionamento dos usuarios através do Microsoft Active Directory, IBM Tivoli Directory Server ou OpenAM data store.

O cliente de cada um dos serviços está disponível como uma API Java. Essas APIs permitem que aplicações e processos consumam os serviços de identidade oferecidos pelo OpenAM. Cada serviço disponibiliza ainda uma interface publica de serviço (SPI) que permite a expansão dos mesmos. Além do SAML 2.0 o OpenAM suporta ainda os protocolos de federação WS-Federation, ID-FF e Liberty ID-WSF.

Os principais componentes da arquitetura do OpenAM são os **SSOTokens**, **agentes** e **fedlets**.

SSOTokens

Usuários podem se autenticar através do uso dos mais diversos tipos de credenciais, como senha, ou certificado. No OpenAM o resultado de uma autenticação bem sucedida é a criação de uma sessão para o usuário/entidade e um Token, o SSOToken, geralmente um número randômico seguro.

O Token é a extrutura básica utilzada como prova da autenticação entre as entidades envolvidas. Como exemplo mais comum podemos utilizar a autenticação de um usuário web. Durante o processo de autenticação um Token é armazenado no navegador web na forma de cookie. Quando o usuário visita outra aplicação web protegida pelo OpenAM o Token é enviado a essa aplicação através do agente instalado no servidor de aplicação . No caso de sucesso o OpenAM então retorna uma cópia da sessão do usuário contendo informações de autenticação, autorização e demais dados como tempo de inatividade através de uma asserção SAML. Caso a requisição não possua o token, o agente redireciona o navegador web para a página de login do provedor de identidade. O token e a sessão serão invalidados quando o usuário executar logout, a sessão expirar, ou ainda um administrador invalidar a sessão do usuário.

Agentes

No OpenAM os agentes, ou *Policy Agents* (PAs), são os responsáveis por proteger recursos web. Baseados no padrão de projeto *interceptor*, agentes interceptam requisições para que possam realizar as checagens de segurança necessárias (autenticação, autorização e auditoria) antes de permitir que o controle seja transferido a aplicação protegida. O uso de agentes permite oferecer recursos como SSO a aplicações que não foram inicialmente projetadas para tal, apesar de que na prática algumas aplicações precisam de mudanças significativas. O OpenAM oferece agentes quer suportam diversos servidores web e de aplicação.

Fedlets

Fedlet é uma implementação leve (*lightweight)* de provedor de serviço utilizando os protocolos SSO do SAML2 embutida em uma aplicação Java EE. Fedlets são parte integrande do OpenAM e podem ser gerados pelo mesmo, possibilitando que aplicações provedoras de serviço simples possam aceitar posts SAML de um Provedor de Identidade. Desta maneira, Provedores de serviço podem participar da federação com esforço mínimo.

3 COMPUTAÇÃO EM NUVEM

3.1 Contexto e Definição

Computação em nuvem é um dos termos emergentes da computação, sendo amplamente discutido e estudado, como também freqüentemente mal interpretado. A Computação em nuvem visa prover acesso sob demanda para um pool de recursos computacionais. Estes recursos podem ser rapidamente providos/liberados com pouco esforço de gerenciamento, pois o ambiente é nativamente dinâmico e facilmente escalável [Mell, Grance 2009].

Apesar de todo cometimento em cima do assunto a idéia de computação em nuvem não é nova, e muito menos nasceu da noite para o dia. Podemos fazer uma interessante analogia do surgimento da computação em nuvem com a revolução industrial [Zhang et al. 2010]. Se olharmos para a história, a revolução aconteceu em estágios, passando por diversos surtos evolutivos que provocaram mudanças graduais, mas drásticas na economia mundial. A internet por sua vez evoluiu de maneira semelhante, gradual e evolutivamente até que as ferramentas necessárias para a nuvem estivessem disponíveis.

A computação em nuvem não é uma nova tecnologia, e sim a junção de tecnologias pré-existentes que amadureceram e evoluíram em contextos diferentes, sem ter como foco a construção de uma nova tecnologia como um todo. Avanços em processadores, armazenamento, tecnologia de virtualização e servidores cada vez mais acessíveis possibilitaram a computação em nuvem emergir como uma realidade viável e competitiva.

A migração de sistemas tradicionais para os serviços fornecidos pela nuvem pretende reduzir os custos de manutenção da infraestrutura de Tecnologia da Informação, oferecendo economia em servidores, armazenamento, rede, licenças de software, energia, resfriamento, redução de trabalho na administração de sistemas e redução do tempo de configuração [Zhang et al. 2010].

3.2 Analogia com o Mundo Real

A idéia de computação em nuvem está intimamente ligada com a idéia de terceirização de serviços. Pegaremos como exemplo uma fábrica de sapatos. Para funcionar, a fabrica utiliza energia elétrica como uma de suas matérias primas básicas. A fábrica poderia ter sua própria

área de geração de energia, mas é menos dispendioso e mais eficiente contratar o serviço de uma concessionária que gere e distribua a energia necessária. Desta maneira estaria evitando a contratação de pessoal especializado, custo de aquisição, manutenção e armazenamento de equipamentos necessários à geração de energia, bem como domínio de conhecimentos intrínsecos, como questões ambientais e de segurança. Pode parecer absurdo que hoje em dia uma fábrica, em condições normais, manteria sua própria unidade de geração de energia, mas isto já foi uma realidade não muito distante. Quando paramos para pensar em recursos computacionais esta realidade se torna ainda mais próxima.

Agora imagine se a mesma fábrica decidisse expandir suas vendas através da internet com um site de vendas online, a primeira necessidade seria a compra de computadores e espaço de armazenamento, uma infraestrutura que necessita de espaço físico e consumo de energia. É necessária ainda a aquisição de diversas licenças de software, entre sistemas operacionais, ferramentas de desenvolvimento, e possivelmente, de um sistema de gerenciamento de banco de dados. Como o negócio da empresa não é o desenvolvimento de sites de vendas online seria necessária à contratação de pessoal especializado, como analistas de sistemas, programadores, designers e DBAs. Entretanto, mesmo montando uma equipe básica, uma aplicação web não é algo simples de construir, exigindo um grande conhecimento prévio dos programadores e analistas em diversas áreas, como frameworks web, servidores de aplicação e configurações do ambiente. Isso sem falar nas questões de segurança, a aplicação teria que manter dados importantes sobre clientes, comunicação com bancos e seria a porta da frente da empresa tanto para clientes como para possíveis atacantes.

Foi necessário um grande investimento inicial apenas para um tentativa de abertura do negócio, agora imagine ainda que, já montada toda a infraestrutura, a aplicação fosse um enorme sucesso de vendas e já não agüentasse mais a carga de requisições. Seria necessário um novo investimento em máquinas, armazenamento, banda de rede, espaço e possivelmente de novas licenças e pessoal especializado.

Todas essas preocupações se afastam em muito do negocio principal da empresa, consumindo uma enorme quantidade de recursos além de dificultar uma futura expansão dos negócios. Ao contrário, se a empresa precisasse de mais energia, seria necessário apenas rever seu contrato com a concessionária.

A idéia da computação em nuvem se assemelha a da concessionária de energia, oferecendo o serviço computacional em um sistema utilitário pague-o-quanto-usa. No nosso exemplo, a fábrica

poderia contratar os serviços computacionais necessários, seja software, infraestrutura ou plataforma computacional contratando apenas a quantidade exata que precisa no momento e com capacidade teoricamente infinita de crescimento à medida que fosse necessário.

3.3 Características da Nuvem

A computação em nuvem se define em 5 características chave [Latif et al. 2009]: compartilhamento de recursos, escalabilidade, elasticidade, pague-o-quanto-usa e auto provisionamento de recursos.

Compartilhamento de recursos

Ao contrário de outros modelos computacionais que utilizam recursos dedicados, a computação em nuvem é baseada em um modelo de negócio no qual os recursos são compartilhados, permitindo desta maneira um melhor aproveitamento da estrutura computacional por trás da nuvem.

Escalabilidade

A utilização de recursos é apenas na quantidade necessária. Uma empresa não precisa adquirir servidores de ultima geração esperando mudanças no futuro. Os recursos computacionais e de armazenamento são disponíveis de acordo com a necessidade atual.

Elasticidade

Os usuários podem aumentar ou diminuir os recursos computacionais à medida que necessitarem. Por exemplo, uma aplicação web empresarial com demanda variável de acessos pode ter uma quantidade de recursos maior alocada em horários comercial e liberar os recursos durante a noite.

Pague-o-quanto-usa

Os usuários pagam apenas pelos recursos que usam e somente na hora em que usam. Os serviços podem ser oferecidos pelo provedor de nuvem de maneira não uniforme, horários de grande acesso a uma aplicação web tem um custo maior do que quando a mesma não está sendo utilizada, semelhantemente ao que acontece com o fornecimento

de energia elétrica por uma concessionária ou a contratação de rede 3G baseada pela banda utilizado que é oferecido por algumas operadoras de telefonia celular.

Auto provisionamento de recursos

Recursos podem ser alocados automaticamente à medida que sejam necessários sem necessitar de, por exemplo, constantes mudanças de configuração do administrador para os diversos horários do dia.

O interesse pela computação em nuvem cresce cada vez mais à medida que proporciona acesso a um poder computacional comparado a supercomputadores, tudo por apenas uma fração do preço que seria adquirir e manter toda uma infraestrutura. Ainda mais importante é a possibilidade de crescimento em demanda que a nuvem proporciona, uma vez que a nuvem é o supercomputador.

3.5 Modelos de Serviço na Nuvem

Os modelos de serviço na nuvem são classificados em três categorias: Infraestrutura como Serviço (IaaS), Plataforma como Serviço (PaaS) e Software como Serviço (SaaS).

Infraestrutura como Serviço (IaaS)

O modelo de serviço IaaS fornece formas de provisionar a infraestrutura computacional necessária em um ambiente de virtualização. A nuvem provê processamento, armazenamento, rede e outros recursos computacionais para que o usuário possa instalar aplicações ou até mesmo sistemas operacionais. O usuário não gerencia diretamente a infraestrutura da nuvem, mas pode provisionar recursos e tem acesso a configurações de diversos componentes. O modelo de IaaS oferece grande escalabilidade e flexibilidade da infraestrutura de TI como também acesso a grande poder computacional de maneira mais rápida, dinâmica e barata do que o modelo tradicional de computação.

Plataforma como Serviço (PaaS)

No modelo de serviço PaaS o provedor fornece uma plataforma computacional onde o usuário pode publicar suas aplicações criadas por

linguagens e ferramentas suportadas pelo provedor da nuvem. O usuário não tem controle sobre a infraestrutura computacional mas gerencia as configurações do ambiente necessárias para o funcionamento de suas aplicações.

Software como Serviço (SaaS)

No modelo SaaS o usuário utiliza aplicações rodando na infraestrutura de nuvem do provedor. O pagamento pelo uso pode se dar por meio de uma assinatura ou pelo modelo pague-o-quanto-usa. O usuário tem acesso apenas às configurações da aplicação, sem possuir informações na infraestrutura em que ela roda. O usuário pode acessar as aplicações de qualquer dispositivo autorizado, seus dados e preferências ficam armazenados na nuvem. Uma empresa pode ser beneficiar do modelo SaaS, por exemplo, contratando os softwares que precisa de um provedor na nuvem e pagando um valor periódico pelo uso, evitando assim ter que adquirir todo o hardware, armazenamento e licenças necessárias para executá-los.

3.6 Modelos de Nuvem

Os modelos de Nuvem podem ser classificados em pública, privada e híbrida.

Nuvem Pública

Nesse modelo a infraestrutura da nuvem fica disponível para o público ou para grupos específicos. A nuvem pública é hospedada e gerenciada por uma empresa ou organização que vende seus serviços na nuvem.

Nuvem Privada

A infraestrutura da nuvem é interna e emula a computação em nuvem pública, mas disponível apenas para uma empresa ou organização específica. Esta por sua vez pode ficar a cargo do gerenciamento da nuvem ou terceirizar o serviço para um provedor especializado. Esse tipo de nuvem busca prover alguns dos benefícios da computação em nuvem pública sem ter grande parte dos problemas de segurança e confiabilidade dos dados de uma nuvem deste tipo.

Nuvem Híbrida

Consiste em múltiplas nuvens computacionais internas e ou externas. Funcionam como entidades separadas, mas usam padrões que possibilitam a portabilidade de dados e aplicações.

A figura 3 resume todas as definições apresentadas para o ambiente em nuvem.

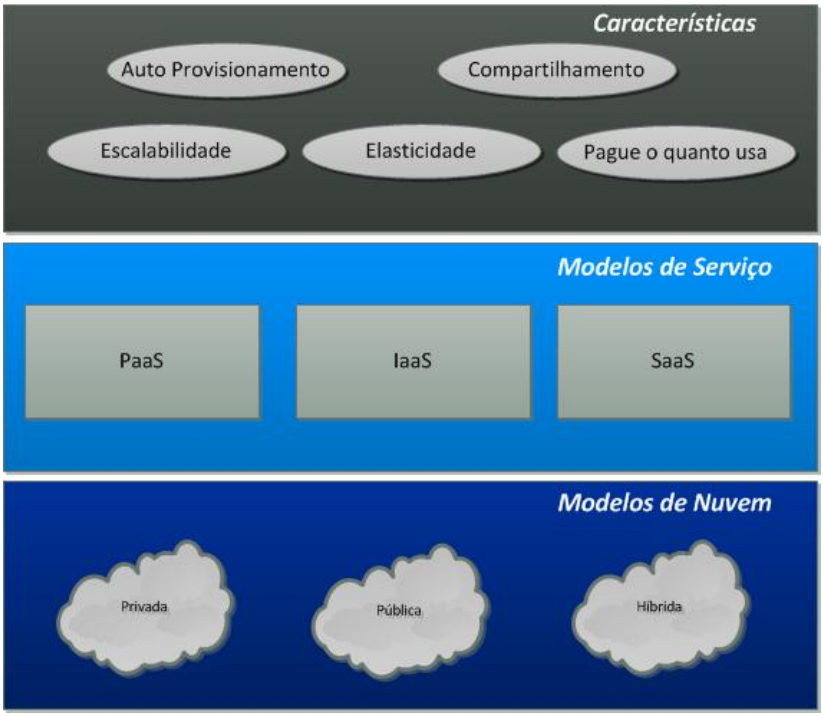

Figura 3: Definições Computação em Nuvem

3.7 Ferramentas Relevantes

3.7.1 Eucalyptus

O Eucalyptus (*Elastic Utility Computing Architecture Linking Your Programs To Useful Systems*) é um software distribuído sobre a licença GPL que facilita a criação e a manutenção de nuvens privadas e publicas no modelo IaaS. O Eucalyptus vem se tornando muito popular e é visto como uma das principais soluções *open source* para plataformas de computação em nuvem.

A nuvem desenvolvida pelo Eucalyptus aborda: agendamento e instanciação de máquinas virtuais (VM - *Virtual Machine*), 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 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 [Marcon et al. 2010].

Eucalyptus é baseado em padrões da indústria, com mecanismos de comunicação independentes de linguagem e estrutura modular. Vem disponível na distribuição Linux Ubuntu desde a versão 9.10 dentro do *Ubuntu Enterprise Cloud* (UEC).

Em uma nuvem Eucalyptus (figura 4) existem quatro componentes de alto nível, cada um com sua interface de serviço web: Node Controler, Cluster Controller, Walrus Storage Controller e Cloud Controller [Nurmi et al. 2009].

Node Controller

O Node Controller é executado em todo nó que hospeda uma maquina virtual, ele consulta o sistemas operacional para obter os recursos físicos disponíveis e controla o ciclo de vida das instancias rodando no nó.

Cluster Controller

O Cluster Controller gerencia um ou mais Node Controllers, geralmente é executado em uma máquina que é a porta de entrada para

o cluster, ou qualquer máquina que possua conectividade de rede com ambos os nós (Node e Cluster Controller). As funções primárias são agendar solicitações para a execução de instâncias em nós específicos, controlar a rede virtual sobreposta e recuperar/enviar informações sobre um conjunto de nós.

Cloud Controller

Os recursos virtualizados que compõem a nuvem Eucalyptus são expostos e gerenciados pelo Cloud Controller. É a fachada para toda a infraestrutura de nuvem disponível pelo Eucalyptus. Tem as funções de monitorar a disponibilidade dos recursos em vários componentes da infraestrutura da nuvem, monitorar as instancias em execução e decidir quais clusters serão usados para provisionar essas instancias.

Walrus Storage Controller

O Walrus Storage Controller provê um simples sistema de armazenamento de dados estendendo APIs padrões na web como o *Simple Object Access Protocol* (SOAP). Tem como funções armazenar as imagens e snapshots das máquinas virtuais e armazenar e servir arquivos usando a API S3 da Amazon.

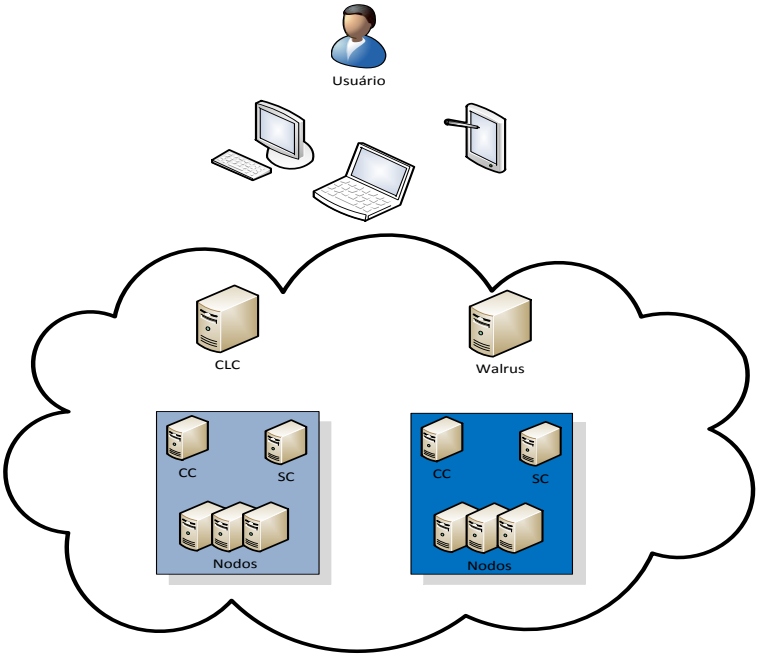

Figura 4: Arquitetura Eucalyptus

3.8 Questões de Segurança na Nuvem

Como explicitado anteriormente, a computação em nuvem é um novo modelo de computação e por este motivo existe um grande grau de incerteza sobre a segurança em todos os níveis. Essa incerteza é bem fundamentada uma vez que a falta de padrões e baixa motivação por parte dos provedores de nuvem em compartilhar detalhes sobre a infraestrutura por trás de seus serviços tornam a migração para o ambiente em nuvem uma jornada perigosa.

A idéia de terceirização por traz da computação em nuvem afeta fortemente a segurança, uma vez que migra grande parte do controle sobre informações críticas da organização para o provedor na nuvem. Como se não fosse o suficiente, processos importantes de segurança na infraestrutura de TI são terceirizados, como a aplicação de atualizações e configuração de firewalls.

Os diferentes modelos de Computação em Nuvem influenciam no controle direto que o cliente tem sobre a infraestrutura de TI e consequentemente na distribuição de responsabilidades na segurança da mesma.

O modelo de Infraestrutura como Serviço (IaaS) é o que menos transfere controle sobre a infraestrutura de TI e responsabilidade de segurança para o provedor de nuvem. Neste modelo o cliente tem controle sobre as configurações do sistema operacional, rede e armazenamento. No outro extremo, o modelo de Software como Serviço (SaaS) é o que mais concentra a responsabilidade no provedor da nuvem, tendo o cliente controle somente sobre configurações do software.

Nem sempre os provedores de serviço na nuvem estão dispostos a compartilhar informações de seus processos de segurança, algoritmos de criptografia, localização geográfica dos centros de armazenamento de dados ou configurações de firewall.

4. Gerenciamento de Identidade como Serviço (IDaaS - *Identity As A Service*)

4.1 Identidade na nuvem

Os objetivos dos métodos de segurança na nuvem não são diferentes dos do modelo tradicional. Um dos parâmetros principais é proteger os dados fundamentais utilizados pelos sistemas e serviços. Neste quesito, as informações de autenticação e identificação dos usuários são de vital importância. À medida que nos movemos para nuvem, os métodos tradicionais de segurança destes dados são desafiados pelas arquiteturas inerentes ao modelo [CSA 2009]. Nos últimos anos a segurança computacional vem convergindo em direção à segurança comercializada como um serviço e neste caso deve-se confiar no vendedor do serviço [Marcon et al. 2010]. Podemos comparar este método com a confiança que necessitamos ter ao fazer um depósito de dinheiro em uma agência bancária. É preciso confiar que o banco esteja íntegro e funcionando corretamente de maneira que o cliente não sofra o risco de perder a quantia depositada.

A segurança e a privacidade são os principais desafios que podem impedir a ampla adoção da abordagem de computação em nuvem, uma vez que falhas de segurança em qualquer um dos componentes podem impactar os demais e, consequentemente, a segurança de todo o sistema será comprometida [CSA 2010] [Hansen et al. 2008].

Para uma organização pode existir um grande potencial de proliferação de identidades e credenciais necessárias para acessar os serviços. Existe a necessidade de limitar corretamente os domínios e limites dos fluxos de informações entre os provedores de serviços na nuvem, uma vez que podem existir fluxos diversos entre vários domínios para que um serviço seja fornecido. Além disso, um provedor de serviço pode ser cliente de outros SPs em outros domínios de segurança que podem até mesmo estarem hospedadas em outras nuvens computacionais.

A propagação das identidades e das informações pessoais (PII) deve ser controlada e monitorada. Isso porque devem ser observadas leis que cumpram a manutenção da privacidade individual. Informações sensíveis como identidades de funcionários, estruturas organizacionais, serviços e aplicações da empresa não podem ser expostas nos

provedores de serviços nas nuvens. É necessário haver controle desses dados [Armbrust et al. 2010] [Hansen et al. 2008].

Delegar os processos de gerenciamento de identidades e de acesso para os provedores de serviço e de computação nas nuvens pode causar riscos: pode não existir a garantia de que esses processos e as práticas de segurança usadas pelos provedores sejam consistentes com as políticas da empresa e pode não existir a garantia sobre a consistência e a integridade das contas e informações de usuários através das nuvens e serviços [Jaeger et al. 2010].

Para que as organizações consumidoras utilizem os serviços oferecidos pela nuvem é necessária a implantação de um modelo de gerenciamento de identidades 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 [Armbrust et al. 2010].

Para o ambiente de nuvem o gerenciamento da autenticação deve fornecer suporte aos processos de criação e emissão das credenciais utilizadas pelos usuários da organização. 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, uma vez que 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 utilizando 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 usuário individual deve ter a oportunidade de consentir ou revogar a liberação de informações pessoais para garantir a sua privacidade. Além disso, o usuário individual quer garantias de que suas informações estão asseguradas contra roubo ou uso indevido [Pearson 2009].

A computação em nuvem permite que os provedores de serviços utilizem a quantidade de recursos necessários para realizar testes com novos sistemas. Se um projeto falhar durante sua fase inicial, por exemplo, o provedor de serviço investiu uma quantia relativamente

pequena no negócio, podendo facilmente alterar suas estratégias ou até mesmo a sua área de negócio. Se a cada nova aplicação fosse necessário desenvolver um modelo de autenticação robusto, confiável e com características necessárias à nuvem, se perderia em muito essa importante característica.

4.2 Características necessárias na Nuvem

O modelo tradicional (descentralizado) de identidades, onde cada aplicação possui seu próprio sistema de autenticação não é recomendável na nuvem, uma vez que os usuários potencialmente se autenticam em diversos serviços e a replicação dos dados de autenticação entre múltiplos domínios com recursos compartilhados não é recomendável em termos de segurança em um sistema distribuído.

Um grande problema do sistema descentralizado é o fato de possuir baixa usabilidade, pois requer que o usuário memorize vários pares usuário-senha e se autentique diversas vezes. Isso é um problema mesmo que o usuário esteja usando apenas uma aplicação, uma vez que na nuvem é comum que a mesma possa acessar diversos outros domínios de segurança para atingir seus objetivos de negócio.

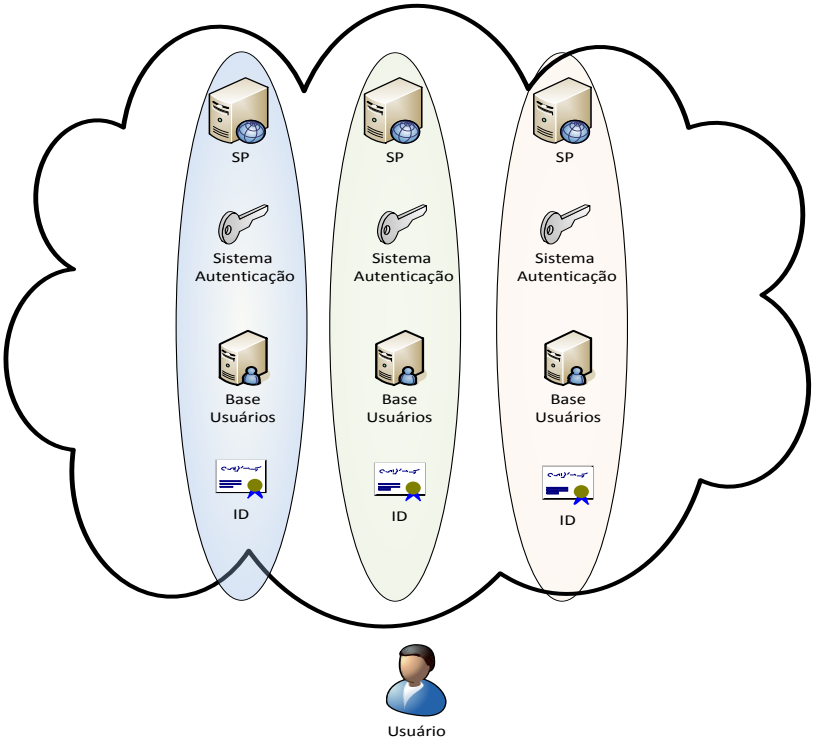

Figura 5: Sistema descentralizado na nuvem

Na figura 5 podemos observar as características deste modelo descentralizado, que força que o usuário se registre em todos os provedores de serviços separadamente, tendo que passar por um processo diferente de cadastro e autenticação em cada um dos recursos acessados. Nesse sistema é altamente provável que as informações pessoais de identificação fiquem desatualizadas, incorretas ou até mesmo que o usuário não seja tão preciso ao informá-las devido a natureza tediosa e repetitiva do processo [Fraga et al. 2010].

O modelo de computação em nuvem necessita de um controle centrado no usuário onde cada requisição a um provedor de serviço seja acompanhada pela identidade do usuário. A identidade deve possuir as informações de autorização e autenticação, bem como informações sobre o usuário que o caracterize (PIIs) e que o mesmo aceite compartilhar com os provedores de serviço [Lee et al. 2009]. O usuário deve poder configurar essas informações de maneira acessível e

centralizada, podendo modificá-las ou revogá-las e as alterações se fazerem válidas para todos os provedores de serviço no mesmo instante.

Os serviços oferecidos na nuvem são heterogêneos e, portanto, se um padrão não for adotado, é altamente provável que utilizem 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. O uso de informações diferentes para montar uma identidade cria ainda 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.

A identidade deve ainda estar ligada a um domínio, mas não limitada a ele, podendo atravessar barreiras entre as diversas aplicações na nuvem e até mesmo fora dos domínios do provedor de nuvem através da federação de identidades.

Como foi abordado no capítulo de gerenciamento de identidades (capítulo 2), muitas destas funcionalidades podem ser alcançadas com o uso de padrões abertos, e já bastante difundidos no modelo de computação tradicional, como o SAML e o OpenID.

Um modelo de gerenciamento de identidade para computação em nuvem deve permitir rápida adaptação, quaisquer mudanças nas informações do usuário devem estar disponíveis imediatamente para todos os domínios. Em um modelo tradicional o gerenciamento de identidade é baseado em um contrato de confiança do usuário com a organização ou domínio. O modelo de nuvem é mais volátil, adicionando desafios como resoluções de domínio dinâmicas e manutenção das informações de segurança ao cruzar domínios desconhecidos.

4.2 Proposta: Gerenciamento de Identidade na Nuvem

Considerando os problemas e requisitos levantados anteriormente, definimos as características necessária ao gerenciamento de identidades para o ambiente de computação em nuvem.

4.2.1 Gerenciamento de Identidades Externalizado como Serviço

Devido à natureza dinâmica dos negócios na nuvem que tem por princípio a rápida disponibilização, o processo de gerenciamento das identidades não deve ser um peso a mais em cada novo serviço na nuvem. Nosso modelo busca um padrão que possibilita externalizar essa carga das aplicações, oferecendo o gerenciamento das identidades na forma de um serviço disponível na nuvem e que possibilite fácil reuso, economia de recursos e que ao mesmo tempo diminui os pontos de falha. Dessa forma podemos aproveitar ao máximo uma importante vantagem da computação em nuvem, a de poder disponibilizar aplicações rapidamente, mesmo quando ainda não se tem certeza do tamanho do ciclo de vida das mesmas e sem ter que montar previamente toda a infraestrutura de TI necessária.

O serviço proposto recebeu o nome de IdaaS (*Identity as a Service* - Gerenciamento de Identidade como Serviço), que funciona como um modelo de gerenciamento de identidade centralizado quando dentro de um mesmo provedor de nuvem (compartilhando portanto a mesma estrutura de nuvem). Como veremos mais adiante a troca de identidades entre nuvens também é possível através do modelo federado.

A figura 6 mostra o IdaaS funcionando de maneira centralizada dentro de uma mesma estrutura de nuvem. O serviço funciona como o provedor de identidades do modelo centralizado e provê o provisionamento dos usuários para os serviços hospedados na nuvem. Os SPs podem ainda compartilhar identidades e informações dos usuários através do IdaaS e tem a possibilidade de login único (SSO) entre seus serviços.

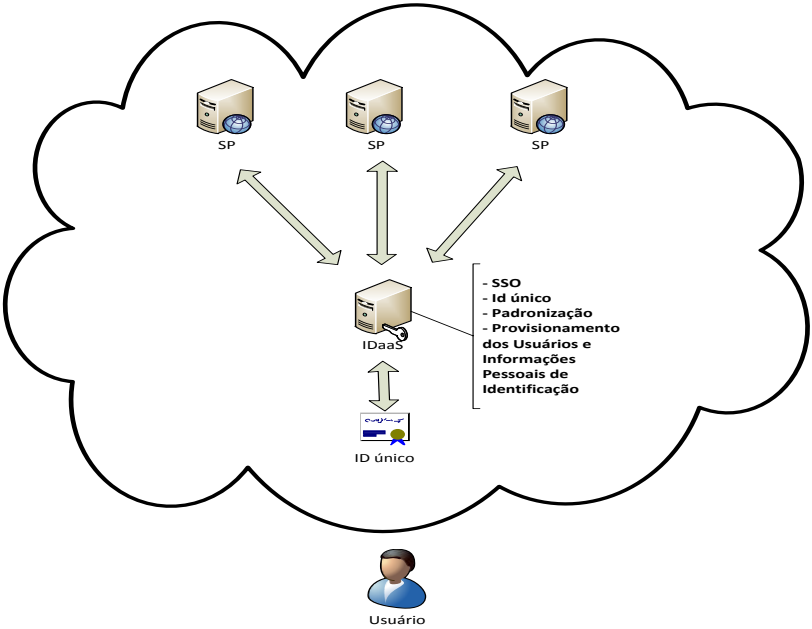

Figura 6: IdaaS como modelo de IDM centralizado em um mesmo CP

O modelo prevê que o serviço seja oferecido e mantido pelo próprio provedor de nuvem (CP - *Cloud Provider*). O IdaaS tem características que julgamos necessárias na nuvem com base nas características citadas na literatura já apresentada.

4.2.2 Provedor de Nuvem como terceira Parte Confiável

Os provedores de serviço (SP – *Service Provider*) precisam trocar informações de autenticação para uma maior usabilidade de seus serviços, disponibilizando vantagens como *Single Sign-On* (SSO) que são essenciais para o ambiente dinâmico e flexível da computação em nuvem. Para que isso seja possível é preciso eleger uma terceira parte confiável que fará autenticação das informações trocadas. No nosso modelo, o próprio provedor de serviços na nuvem foi o eleito, uma vez que tem melhores condições de proteger as informações sensíveis na nuvem e precisa manter um contrato formal com todos os provedores de serviço que utilizam sua estrutura para disponibilizar seus serviços. Os

provedores de serviço que desejarem utilizar o gerenciamento de identidades disponibilizado pelo provedor de nuvem podem incluí-lo ainda no acordo de nível de serviço (SLA) que mantém com o mesmo.

4.2.3 Provisionamento dos usuários pelo Provedor de Nuvem

Buscando a centralização, padronização e segurança dos usuários e informações pessoais de identificação, os mesmos são provisionados pelo IdaaS.

Existe uma grande preocupação com a proteção física de dados sensíveis armazenados na nuvem. O provedor na nuvem pode ter centros de armazenamento de dados espalhados em vários lugares do mundo e muitos destes provavelmente têm no preço o fator preponderante no momento da implantação. O provedor de nuvem sendo o responsável pelo armazenamento dos usuários e PIIs tem a possibilidade de selecionar apenas locais fisicamente seguros para o armazenamento destas informações.

4.2.4 SSO

Como o modelo de negócio na nuvem precisa ser dinâmico e flexível devido a sua própria natureza, um sistema de autenticação único e centralizado se torna necessário. Aplicações parceiras precisam de um meio fácil e acessível de autenticação e troca de atributos sem ter que forçar o usuário a executar um processo de autenticação toda vez que um provedor de serviço diferente for acessado. Nosso modelo proporciona o login único entre os provedores de serviço hospedados na mesma estrutura de nuvem, onde todos os SPs confiam no IDaaS que é disponibilizado na estrutura como um serviço pelo provedor de nuvem.

4.2.5 Federação de identidades entre Provedores de Nuvem

O acesso por login único (*Single Sign-On* - SSO) não deve estar limitado apenas aos provedores de serviços dentro de uma mesma nuvem. É desejável que aplicações parceiras possam confiar em informações provenientes de provedores de serviço hospedados em outras nuvens. Dessa maneira, além de aumentar a usabilidade dos serviços e flexibilidade dos negócios, os provedores de serviço (SPs)

têm uma liberdade muito maior para mudar de provedor de nuvem caso seja necessário ou conveniente, não se limitando a ter que utilizar o mesmo provedor de nuvem em que estão hospedados seus parceiros. Para este objetivo estabelecemos redes de confiança entre os IDaaS nas diferentes nuvens. Essa característica está ilustrada na figura 7.

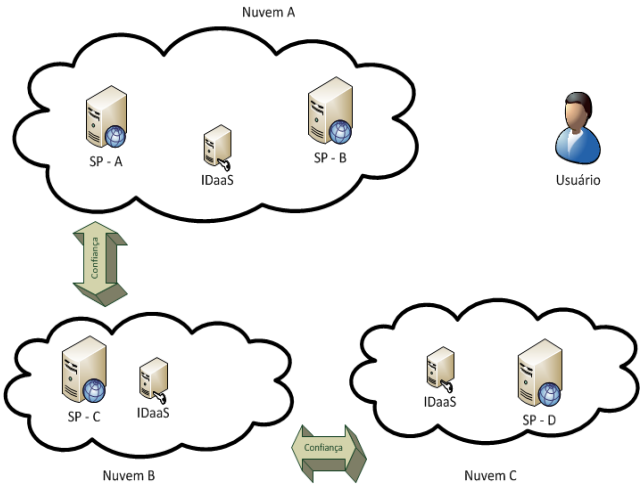

Figura 7: Circulo de Confiança entre nuvens

A fim de que provedores de serviço da nuvem A solicite recursos para a nuvem B, o IDaaS da nuvem A precisa criar uma conta no IDaaS da nuvem B estabelecendo a relação de confiança. No cenário representado, o IDaaS da nuvem B também possui uma relação de confiança com a Nuvem C. Um possível cenário de acesso seria o usuário se autenticando no IDaaS da nuvem A e em seguida realiza o acesso a recursos na nuvem B enviando o token de segurança. O recurso verifica a validade do mesmo enviando um pedido de validação ao IDaaS de sua nuvem (B). Como os IDaaS do cliente e do serviço possuem uma relação de confiança a validação terá sucesso e o SP terá como decidir o acesso com base nas informações enviadas.

4.2.6 Controle das identidades centrado ao usuário e controle das PIIs compartilhadas com os provedores de serviços

A privacidade das informações de identificação pessoais (PII) é uma das principais preocupações dos usuários na adoção a nuvem [CSA 2010] [Pearson 2009]. O usuário precisa ter controle das suas informações que são armazenadas na nuvem, como também o controle de quais informações serão disponibilizadas aos provedores de serviço. O modelo de armazenamento centralizado disponibilizado pelo IdaaS facilita o controle destas informações.

O controle de quais informações serão compartilhadas com os SPs não pode ser totalmente transparente ao usuário. De fato, o sistema deve prover ao usuário controle sobre essas informações, uma vez que informações sensíveis que ele queira disponibilizar a um SP podem não ser factíveis de compartilhar com outro SP com fins diversos. O usuário precisa ter não só o controle das informações compartilhadas, mas também controlar com que provedores estas informações serão compartilhadas, aumentando assim seu grau de confiança em adotar serviços na nuvem. Este controle gera um passo adicional ao acesso aos recursos.

No modelo, o próprio usuário cria e mantém sua conta no IdaaS para poder utilizar os serviços oferecidos pelos SPs hospedados na nuvem e que fazem parte do círculo de confiança. Durante o processo de cadastro o usuário informa ainda suas informações pessoais de identificação. Essas informações não são compartilhadas por padrão com os provedores de serviço e caso os mesmos desejem acessá-las, após o processo de identificação, é feito um novo redirecionamento ao IdaaS para que o usuário seja questionado se deseja liberar o acesso as informações para o provedor de serviço.

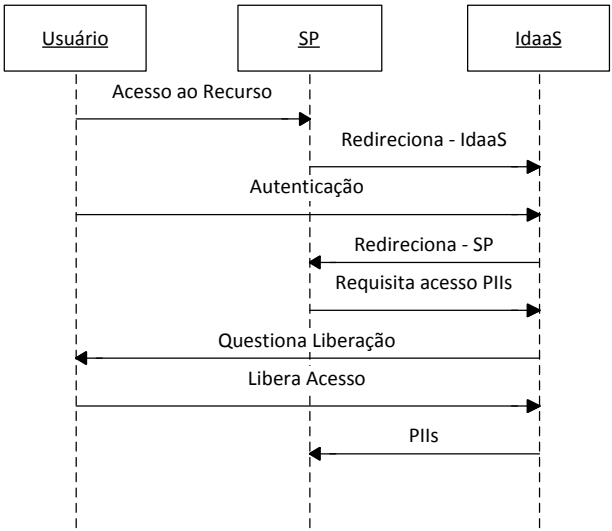

Figura 8: Requisição acesso as PIIs

O funcionamento do processo de requisição, quando o usuário ainda não está autenticado, pode ser observado na figura 8. No passo 1 o usuário acessa o recurso. Como ele ainda não está autenticado, ou seja , não possui o token de segurança e nem um contexto de segurança no IdaaS, é redirecionado para a página de login do IdaaS no passo 2. O passo 3 é o processo de autenticação com o IdaaS. Após autenticado o SP em questão requisita as informações pessoais de identificação do usuário. O usuário é então redirecionado para a pagina do IdaaS que o questiona sobre a liberação do nível requerido das informações pessoais de identificação para o SP em questão. O usuário confirma a liberação e o SP recebe as informações do IdaaS.

4.2.7 Níveis de liberação das Informações Pessoais de Identificação

Nem todos os provedores de serviço precisam de todas as informações do usuário para permitir o acesso a seus serviços. Da mesma forma, os usuários não devem liberar todas as suas informações a todos os provedores, elas devem ser disponibilizadas sempre o mais

próximo do mínimo o possível. Para tal o modelo prevê a criação de níveis de PIIs.

Uma limitação é que um padrão precisa ser adotado, ou seja, todos os provedores de serviço devem utilizar as mesmas informações e nos mesmos níveis. É o provedor de nuvem que define que informações serão utilizadas e em quais níveis elas estarão. Isto é configurado quando da implantação do IdaaS na infraestrutura de nuvem e todos os SPs que desejem utilizar o modelo devem estar conscientes da disponibilidade de tais informações.

Quando o usuário requisita acessar um recurso de um SP o mesmo informa ao IdaaS que nível de informações de autenticação e autorização necessita. O usuário então é redirecionado a página de liberação de acesso e questionado se deseja compartilhar essas informações com o provedor de serviços.

5 Validação

5.1 Ferramentas Utilizadas

Para a implementação da validação do IdaaS foi utilizada a linguagem Java e tomado como ponto de partida o código do projeto open source para gerenciamento de identidades OpenAM 9.5.1. Utilizamos suas capacidades de troca de informações de autenticação através do SAML 2.0 e a capacidade de gerenciar círculos de confiança. Customizamos fortemente sua estrutura desenvolvendo as funcionalidades necessárias ao funcionamento do modelo e removendo os módulos que não eram necessárias para o nosso modelo na nuvem.

O IdaaS é distribuído na forma de um arquivo .WAR que contém as paginas web (configuração, controle de usuários, autenticação, etc) e as interfaces e serviços Java, e que pode ser implantado em um servidor de aplicação com suporte a JavaEE, sendo o Glassfish o servidor de aplicação utilizado para o desenvolvimento e testes e o único que teve o agente do OpenAM configurado para o correto funcionamento do modelo.

Para o ambiente de teste foi utilizada o Ubuntu Server Edition 10.10 para o Cloud Cluster e para os nós da infraestrutura da nuvem (Node Controller), utilizando o suporte por padrão no sistema do Ubuntu Enterprise Cloud (UEC), que é baseado no Eucalyptus, para controle da infraestrutura (modelo IaaS da nuvem).

Para gerenciar as instâncias no UEC foi utilizado o plugin do Firefox HybridFox .

Ubuntu 10.04 para a instância do SO rodando na infraestrutura virtualizada no UEC e o servidor de Aplicacao GlassFish Project - V2 UR2 Final Build para deploy do IdaaS e provedores de serviço.

5.2 Implantando o ambiente de testes

Para iniciar os testes foi necessário primeiramente montar duas infraestruturas de nuvens privadas, nas quais foram implantadas o IDaaS e os provedores de serviço. Mostraremos na próxima sessão como montar uma infraestrutura de nuvem privada utilizadando o Ubuntu Enterprise Cloud que é basicamente uma versão do Eucalyptus com um processo de configuração ligeiramente facilitado.

5.2.1 Processo de Instalação Ubuntu Enterprise Cloud

A instalação do ubuntu server não é muito diferente de uma instalação normal do ubuntu. Cobriremos apenas os passos adicionais de uma instalação com o Ubuntu Enterprise Cloud.

O primeiro passo é fazer o boot da mídia de instalação do Ubuntu Server. No inicio do processo de instalação será apresentada as opções de instalação como observado na figura 9. Escolha a opção "Instalar Ubuntu Enterprise Cloud".

Figura 9: Tela inicial processo de instalação UEC

O instalador irá detectar automaticamente se existem outros componentes da infraestrutura da nuvem (Cloud Cluster, Node Controler, Walrus) e caso nenhum Cloud Cluster seja acessível, é requisitado para que um seja configurado. Esse processo pode ser observado na figura 10.

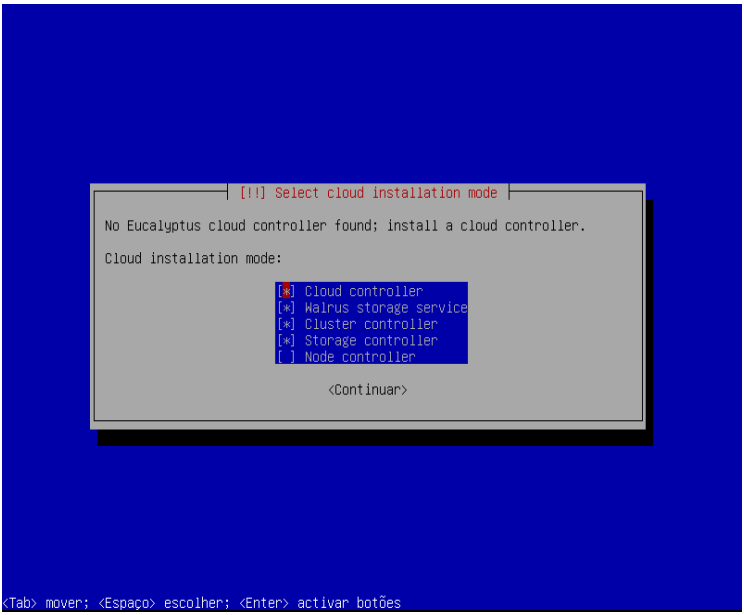

Figura 10: Seleção Modo de Instalação

O instalador irá solicitar um intervalo de endereços IP públicos na LAN que precisam ser definidos para que a nuvem possa atribuí-los as instâncias (ex. 192.168.1.200-192.168.1.249).

Caso o hardware não suporte virtualização por hardware, uma tela como a figura 11 é apresentada, indicando que o Eucalyptus irá rodar de maneira limitada.

O processo de instalação dos nós é bastante semelhante. Basta repetir o processo visto até aqui em cada computador que servirá como nó da infraestrutura de nuvem. É importante notar que é preciso que os nós estejam na mesma rede que o Cloud Controller para que os mesmos possam ser registrados automaticamente pelo instalador.

Figura 11: Processador sem suporte a virtualização por hardware

Finalmente, após a instalação é possível acessar a interface web do Cloud Controler através do endereço http://<ip_Cloud_Cluster>: 8443/ de qualquer computador que tenha acesso pela rede.

A figura 12 mostra a tela de login, sendo que o usuário padrão criado pelo processo de instalação é “admin” e senha “admin”. Após o primeiro acesso é possível modificá-lo caso seja necessário.

A interface web nos permite instalar nossas próprias imagens dos sistemas operacionais que irão rodar sobre nossa infraestrutura de nuvem, como também oferece uma série de opções de imagens através de uma espécie de “loja” na interface web. Utilizamos uma das imagens disponibilizadas a Ubuntu 10.04 LTS – Lucid Lynx (i386). O processo de instalação é completamente automático e pode ser observado na figura 13.

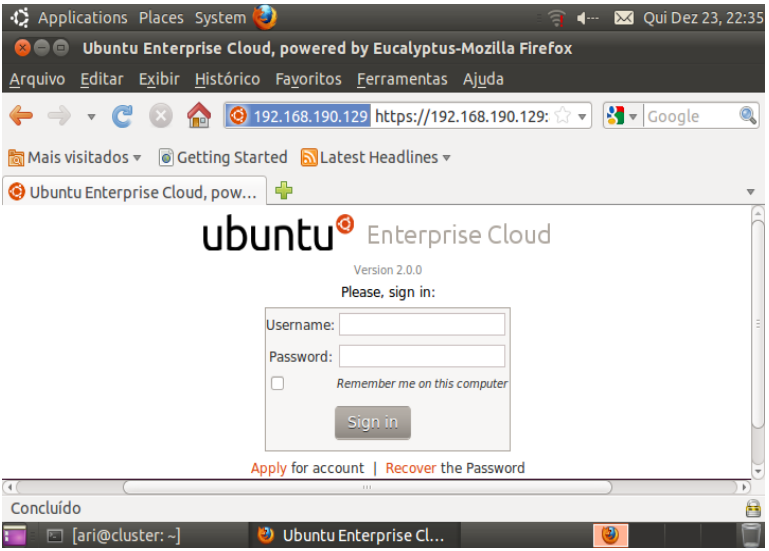

Figura 12: Tela de acesso UEC

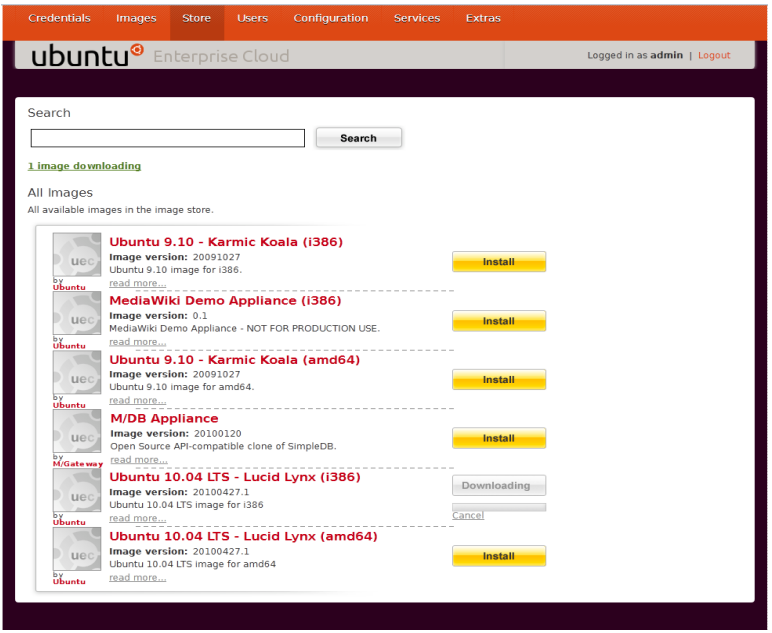

Figura 13: Tela de seleção e instalação de imagens

5.2.2 Gerenciando as Instâncias na Nuvem

Após a instalação da imagem temos nossa infraestrutura de nuvem pronta. Para gerenciar as instâncias e poder então inicializar o SO instalado, optamos pelo uso do HybridFox, um plugin do Firefox que busca gerenciar os dois ambientes de computação em nuvem mais populares o Amazon EC2 e o Eucalyptus. Criado a partir do código fonte de outro plugin de código aberto, o ElasticFox, possui funcionalidades semelhantes com o adicional do suporte ao Eucalyptus.

O HybridFox utiliza o conceito de regiões e precisamos criar uma apontando para o nosso Cloud Controller recém criado. Para fazer isso temos que seguir os seguintes passos:

1. Clique no botão "Regions"
2. Coloque um identificador em "Region Name"
3. Em "Endpoint URL" insira:
   http://<ip_CloudCluster>:8773/services/Eucalyptus
4. Clique em add

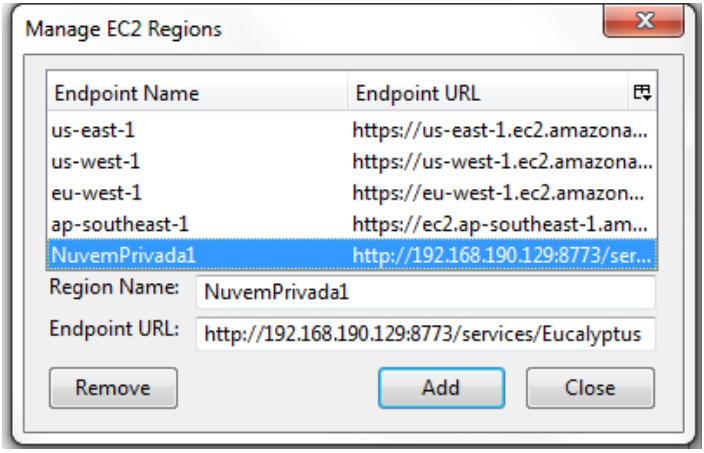

Figura 14: Configurando a url da nuvem privada no HybridFox

O Próximo passo é acessar o Cloud Controler e em credenciais copiar os campos “Query Id” e “Secret Key”, como mostra a figura 15.

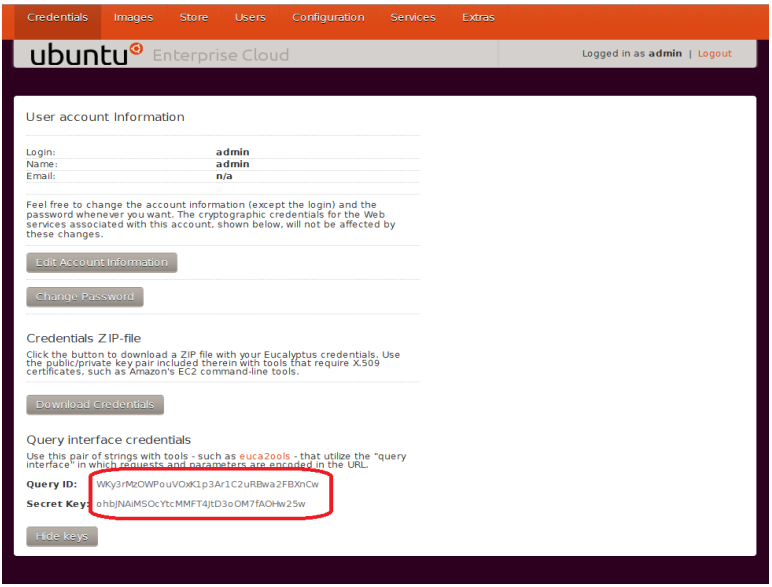

Figura 15: Credenciais

E adicioná-los no HybridFox conforme a figura 16.

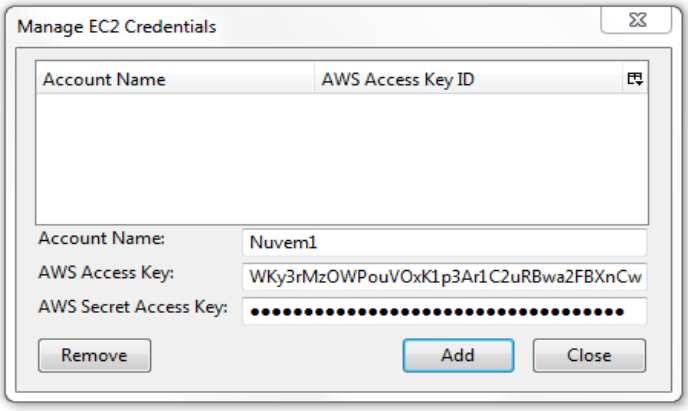

Figura 16: Inserindo Credenciais

Após esse processo o HybridFox estará completamente funcional e será possível iniciar as instâncias instaladas. As instancias iniciadas podem ser acessadas por meio de SSH para o IP atribuído a mesma.

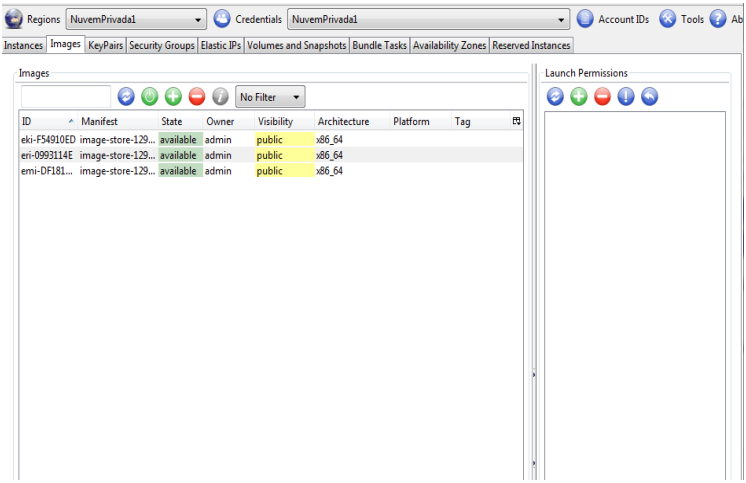

Figura 17: HybridFox Configurado

5.3 Implantação e Funcionamento

O IdaaS é distribuído na forma de um arquivo .War que pode ser implantado em um servidor de aplicação com suporte a JavaEE, utilizamos o para tal GlassFish Project - V2 UR2 Final Build com o seu respectivo agente do OpenAM para interceptação das requisições e proteção dos recursos no servidor. A infraestrutura criada pode ser visualizada nas figuras 17 e 18 e o hardware utilizado está descrito nas tabelas 1 e 2. Foram montadas duas nuvens privadas com seu próprio IdaaS.

Para provedores de serviço foram utilizadas 3 aplicações web simples que tentam acessar as informações pessoais dos usuários. A distribuição das aplicações pode ser observada nas figuras 18 e 19.

Nuvem 1
Cluster e Nó
Model : Toshiba Satellite A665
Processor
Model : Intel(R) Core(TM) i7 CPU Q 740 @ 1.73GHz
Speed : 1.86GHz
Mainboard : Toshiba NWQAA
Total Memory : 4GB DDR3 SO-DIMM
Chipset
Model : Intel Core (Clarksfield/Lynnfield) DMI
Front Side Bus Speed : 2x 2.4GHz (4.79GHz)
Total Memory : 4GB DDR3 SO-DIMM
Channels : 2
Memory Bus Speed : 2x 532MHz (1GHz)
Video System
Video Adapter : NVIDIA GeForce GT 330M
Storage Devices
TOSHIBA MK6465GSX (640.1GB, SATA300, 2.5", 5400rpm, 8MB Cache)
Nó
Model : Sony VGN-NW130J
Processor
Model : Intel(R) Core(TM)2 Duo CPU P7570 @ 2.26GHz
Platform Compliance : Intel Centrino Duo Mobile Technology
Mainboard : Sony VAIO
Total Memory : 4GB DDR3 SO-DIMM
Chipset
Model : Sony PM45 Mobile Controller Hub
Front Side Bus Speed : 4x 199MHz (796MHz)
Total Memory : 4GB DDR3 SO-DIMM
Channels : 2
Memory Bus Speed : 2x 398MHz (796MHz)
Video System
Video Adapter : ATI Mobility Radeon HD 4500/5100 Series
Storage Devices
WDC WD3200BEVS-26VAT0 (320GB, SATA150, 2.5", 5400rpm, 8MB Cache)

Tabela 1: Hardware Nuvem 1

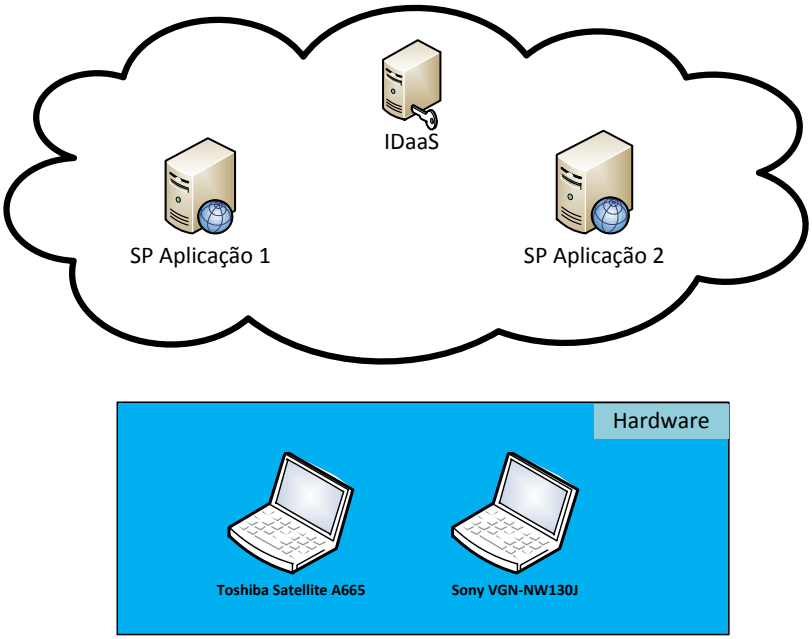

Figura 18: Estrutura Nuvem 1

Nuvem 2
Cluster e Nó
Model : ECS A780GM-A
Processor
Model : AMD Phenom(tm) 9550 Quad-Core Processor
Speed : 2.2GHz
Mainboard : ECS A780GM-A
Total Memory : 3.75GB DIMM DDR2
Chipset
Model : ECS RS780/880 Host Bridge
Front Side Bus Speed : 2x 1.8GHz (3.6GHz)
Shared Memory : 256MB
Channels : 1
Memory Bus Speed : 2x 100MHz (200MHz)
Video System
Video Adapter : ATI Radeon HD 3200 Graphics (40 SM4.0, 256MB, Integrated Graphics)
Storage Devices
Seagate ST3500630AS (500.1GB, SATA300, 3.5", 16MB Cache) : 466GB
Nó
Model : POSITIVO POS-PQ35AS
Processor
Model : Pentium(R) Dual-Core CPU E5300 @ 2.60GHz
Speed : 2.6GHz
Mainboard : Positivo Informatica SA POS-PQ35AS
Total Memory : 4GB DIMM DDR2
Chipset
Model : ASUS Q35 DRAM Controller
Front Side Bus Speed : 4x 200MHz (800MHz)
Total Memory : 4GB DIMM DDR2
Channels : 2
Memory Bus Speed : 2x 250MHz (500MHz)
Video System
Video Adapter : Intel(R) Q35 Express Chipset Family (PS2.0, PCI)
Storage Devices
Seagate ST3160813AS (160GB, SATA300, 3.5", 7200rpm, 8MB Cache) : 149GB

Tabela 2: Hardware Nuvem 2

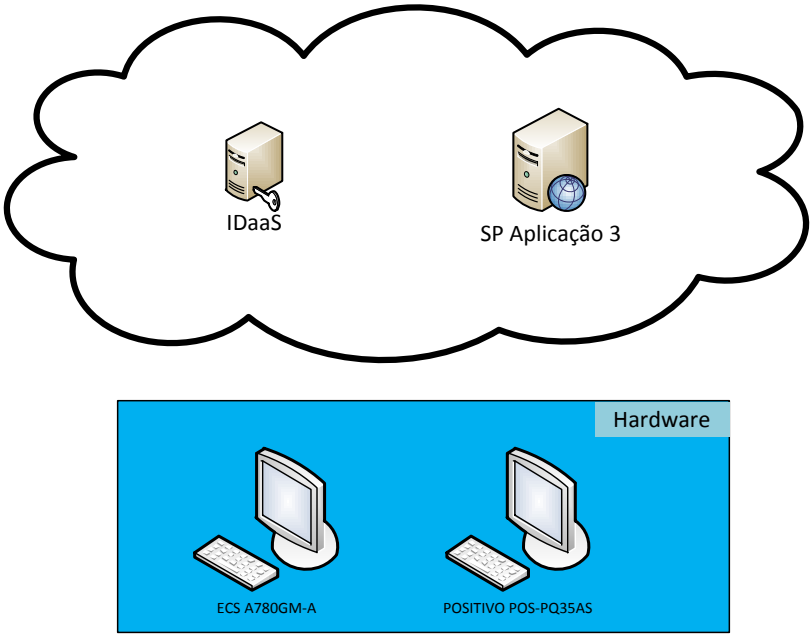

Figura 19: Extrutura Nuvem 2

5.3.1 Registro dos SPs no IDaaS

Primeiramente é necessário que os SPs que desejam utilizar o gerenciamento de identidades disponibilizado pela nuvem efetuem o cadastro no IDaaS. Estes provedores de serviços cadastrados também formam um círculo de confiança entre eles, uma vez que existe a possibilidade de autenticação única entre seus serviços, possibilitado pela confiança mútua que o mesmos depositam no IDaaS. Para isso foi utilizado o conceito do OpenAM de agentes (*Policy Agents*) que protegem as aplicações hospedadas no servidor redirecionando as páginas protegidas para a página de login do IdaaS. O OpenAM possui uma interface web que possibilita o registro de SPs remotos como pode ser observado na figura 20.

Figura 20: Tela de registro de SPs OpenAM

5.3.2 Cadastro dos usuários

Os usuários que desejam utilizar os serviços dos SPs que fazem parte do círculo de confiança da nuvem devem se cadastrar através do IDaaS. Os usuários cadastrados terão acesso aos serviços de todos os SPs que confiem no IDaaS. Como é um modelo centrado ao usuário, o próprio deve criar e manter seus dados atualizados no IDaaS para utilizar os serviços. O usuário acessa, através do navegador, o portal do IdaaS disponibilizado pelo provedor de nuvem e passa pelo processo de cadastro de usuários como pode ser observado na figura 20.

5.3.3 Cadastro e manutenção dos usuários e PIIs

Os usuários devem cadastrar suas informações entrando no portal disponibilizado pelo IdaaS. Nesse portal o usuário pode verificar seus dados e modificar seus dados compartilhados com os provedores de serviços.

Para nível de validação consideramos apenas 2 níveis de PIIs que os usuários podem compartilhar com os provedores de serviços.

O próprio OpenAM possui uma interface de configuração para os atributos compartilhados pelos SPs dentro de um círculo de confiança que tem o intuito apenas de demonstrar as funcionalidades e permitir provas de conceito de maneira rápida. Para criar um padrão dentro do círculo de confiança e facilmente configurar uma nova nuvem estendemos essa funcionalidade para os atributos utilizados na validação.

A tabela 3 mostra essas informações e como elas estão separadas nos níveis e a figura 22 a tela de cadastro de usuários.

Primeiro Nível	
pn	Primeiro Nome
rn	Restante do nome
Sexo	Sexo do usuário
Segundo Nível	
email	Endereço de email do usuário
CPF	número cpf do usuário
telefone	número telefone para contato
CEP	número CEP do usuário

Tabela 3: PIIs Compartilhadas e Níveis de Acesso

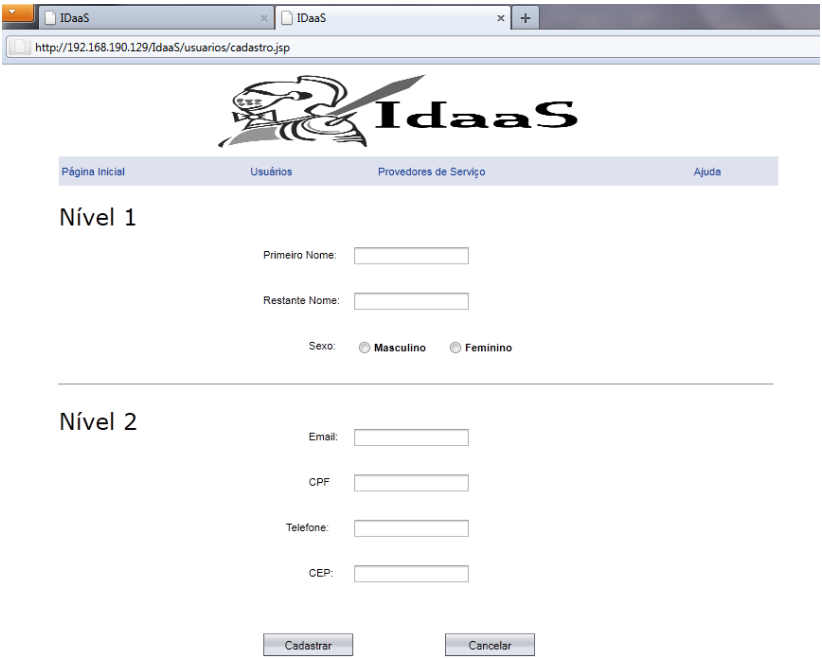

Figura 21: Tela Cadastro PIIs IdaaS

5.3.4 Funcionamento do processo de autenticação

O funcionamento do processo de autenticação ao acessar um SP hospedado na nuvem pela primeira vez em uma sessão pode ser observado na figura 21. Para tal, usamos o *profile* SSO do SAML (através dos serviços do OpenAM), sendo que, no nosso modelo, o IdaaS funciona como o provedor de identidades para os provedores de serviços hospedados na nuvem que queiram participar da rede de confiança.

1. O usuário, por meio do navegador, tenta acessar um recurso de um SP através da URL referente ao recurso.
2. Como o usuário ainda não está autenticado, o servidor web redireciona o usuário para a página de autenticação do IDaaS e gera uma mensagem AuthnRequest (do protocolo de pedido de

autenticação do SAML) e passa para o usuário, para que este apresente ao IDaaS.

3. No passo 3, o navegador entra em contato com o IDaaS para que este autentique o sujeito usando a requisição emitida anteriormente. Nessa etapa é mostrada a tela de autenticação do IDaaS como mostrado na figura 23: o usuário se autentica e o IDaaS inicia o contexto de segurança para aquele usuário e gera as asserções SAML para enviar ao SP. Caso o usuário já tenha se autenticado previamente no IDaaS e já possua um contexto de segurança válido, o passo de autenticação do usuário é omitido.
4. No passo 4, o IDaaS envia a mensagem *Response* do SAML para o SP, por intermédio do navegador utilizando um token de segurança.
5. No passo 5, com base nas informações contidas na resposta o SP inicia um contexto de segurança.

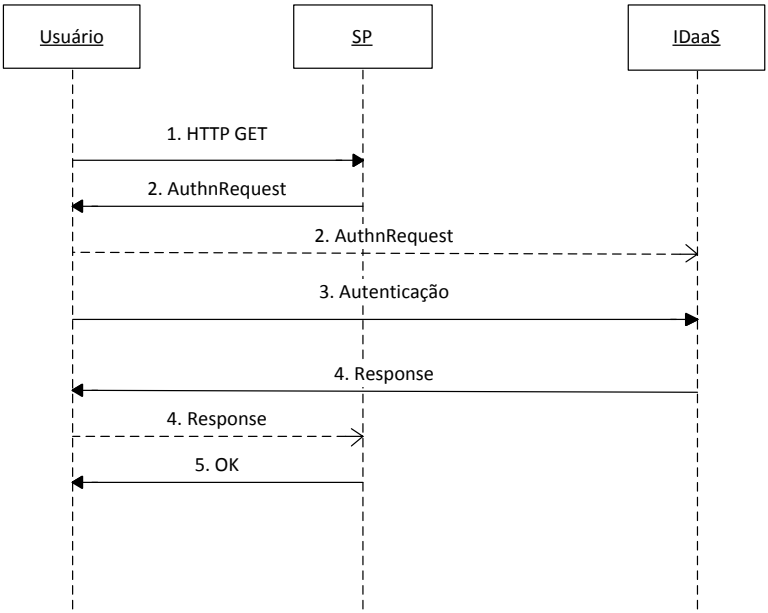

Figura 22: Funcionamento do processo de SSO utilizando o IDaaS

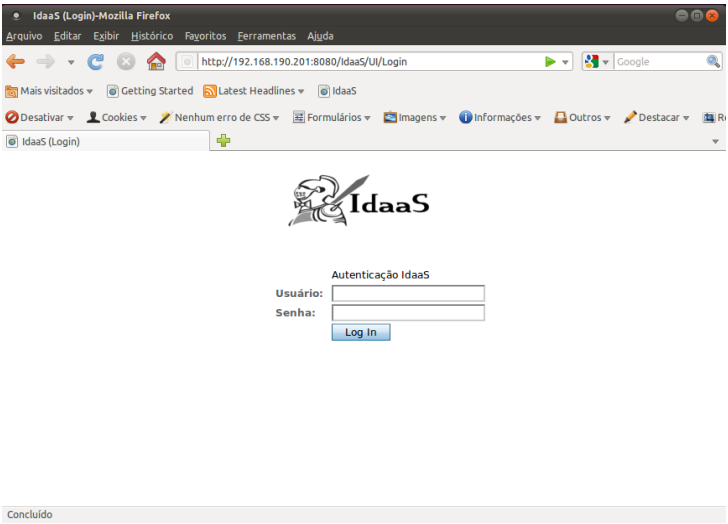

Figura 23: Tela autenticação IdaaS

5.3.5 Liberação das PIIs

Após o processo de autenticação o SP pode requerer as informações pessoais de identificação do usuário ao IdaaS. O navegador é então redirecionado para a página de liberação do IdaaS (figura 24) e o usuário é questionado se deseja liberar as informações pessoais de identificação requisitadas pelo SP que está tentando acessar. Em sua versão atual, o protótipo tem suporte a apenas a 2 níveis de informações pessoais de identificação.

Figura 24: Liberação das PIIs

Figura 25: Aplicação 2 com acesso as PIIs

6. Conclusão e Trabalhos Futuros

Apesar dos grandes avanços nos últimos anos em protocolos e ferramentas para o gerenciamento de identidades, quando o contexto é levado para a nuvem, torna-se um enorme desafio tanto para usuários quanto aos provedores de serviço e de nuvem. Nesta vertente, disponibilizar o gerenciamento destas identidades como serviço na nuvem, disponível como padrão e com esforços centralizados de técnicas de segurança para proteção das informações tem se mostrado o caminho mais natural.

Soluções práticas para o gerenciamento de identidades quando da adoção a nuvem ainda estão em fase inicial de busca de soluções quando observamos a comunidade cientifica e literatura disponível. Encontramos em sua maioria modelos teóricos, novos conceitos e nomenclaturas [Celesti et al. 2010] problemas e recomendações [CSA 2010]. Fortes iniciativas nesta área aparecem com intensidade apenas no meio privado, grandes empresas como HP [Mont 2010] e Oracle [Kaushik 2009] têm apostado nesta área e anunciado o desenvolvimento de soluções comercias, algumas das quais já se encontram em fase inicial de implantação. Este trabalho buscou preencher essa lacuna ao propor um modelo baseado em padrões abertos e soluções adotadas para o gerenciamento de identidades nos últimos anos no modelo tradicional e a utilização e customização de soluções open source para seu desenvolvimento. Podendo assim servir de passo inicial para pesquisadores que busquem desenvolver soluções práticas na nuvem mas que encontram dificuldade em iniciar os trabalhos pela falta de material direcional na área.

A aplicação proposta de gerenciamento de identidade não é viável por padrão para todos os tipos de aplicações que podem ser hospedadas na nuvem. Por exemplo, em modelos de nuvens privadas internas de uma organização, pode não ser desejável que o próprio usuário mantenha suas informações. Em sistemas legados, pode ser totalmente inviável a adoção de um padrão sem que mudanças sensíveis sejam executadas nos aplicativos. Por ser um modelo orientado a serviço a sua adoção é mais natural em sistemas que já são orientados a serviço por natureza. Sistemas legados e fortemente acoplados necessitam de sistemas de adoção mais flexíveis, para que possam entrar no círculo de confiança e trocar informações de autenticação, sem que a aplicação tenha que ser totalmente reformulada para tal.

O protótipo construído para prova de conceito possui funcionalidades simples e não está preparado para uma aplicação em

mundo real. É necessário definir um padrão para as informações pessoais compartilhadas e em que níveis de segurança elas estarão agrupadas de forma que atenda a maior gama possível de necessidades dos provedores de serviços.

Como trabalhos futuros é preciso expandir e definir padrões para as informações dos usuários compartilhadas e buscar métodos que protejam essas informações não apenas dos provedores de serviço, mas também dos provedores de nuvem. Métodos de proteção dos dados dos usuários contra acesso não autorizado pelo provedor de nuvem, como cifrar as informações armazenadas utilizando chaves fornecidas pelo usuário, podem diminuir o nível de confiança que o usuário necessita depositar no provedor de nuvem.

O próximo passo é a melhoria do modelo com o suporte a diversos níveis de autenticação variando com o método utilizado possibilitando um sistema de classificação dos métodos de acordo com a segurança da autenticação, em diferentes níveis de aceitação. Essas informações poderão ser requeridas pelos provedores de serviço para decidir o acesso a recursos com base na segurança do nível de autenticação do usuário. Por exemplo, alguns recursos especializados podem não aceitar autenticação feita por meio de um par usuário-senha, negando assim o acesso.

REFERÊNCIAS

[Armbrust et al. 2010] ARMBRUST, Michael et al. A View of Cloud Computing. *Communications of the ACM*, v. 53, n. 4, 2010. p. 50-58.

[Bhargav-Spantzel et al. 2007] BHARGAV-SPANTZEL, A. et al. User centricity: a taxonomy and open issues. *Journal of Computer Security*, v. 15, n. 5, 2007, p. 493–527.

[Buyya et al. 2008] BUYYA, R.; YEO, Chee Shin; VENUGOPAL, S. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. In: CONFERENCE ON HIGH PERFORMANCE COMPUTING AND COMMUNICATIONS. 10. *Anais...* IEEE Computer Society, 2008, p. 5-13.

[Carmody et al. 2005] CARMODY, S. et al. *Incommon technical requirements and information*. 2005.

[Celesti et al. 2010] CELESTI, Antonio et al. Security and Cloud Computing: InterCloud Identity Management Infrastructure. In: WORKSHOPS ON ENABLING TECHNOLOGIES: infrastructure for collaborative enterprises. 2010. *Anais eletrônicos...* Disponível em: <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5541619>. Acesso em: 12 dez. 2010.

[Chadwick 2009] CHADWICK, D. Federated identity management. *Foundations of Security Analysis and Design V*, 2009. p. 96–120.

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

[CSA 2010] CSA. Domain 12: Guidance for identity & access management v2.1. 2010.

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

[Eve; Drummond 2008] EVE, Maler; DRUMMOND, Reed. The Venn of Identity: Options and Issues in Federated Identity Management, *IEEE Security and Privacy*, v. 6, n. 2, p. 16-23, mar./abr. 2008

[Fraga et al. 2010] FRAGA, Joni da Silva et al. Gerenciamento de identidades federadas. In: SIMPÓSIO BRASILEIRO EM SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS, 10., 2010, Fortaleza. *Anais eletrônicos***...** Fortaleza: SBSeg, 2010. Minicurso. Disponível em: <http://www.insert.uece.br/sbseg2010/anais/04_minicursos/minicurso_01.pdf>. Acesso em: 01 dez. 2010.

[Hansen et al. 2008] HANSEN, Marit; SCHWARTZ, Ari; COOPER, Alissa. Privacy and Identity Management. *IEEE Security and Privacy*, v. 6, n. 2, p. 38-45, mar/abr, 2008.

[Huang et al. 2010] HUANG, He Yuan et al. Identity Federation Broker for Service Cloud. In: INTERNATIONAL CONFERENCE ON SERVICE SCIENCES. 2010. *Anais eletrônicos...* Disponível em:

<http://ieeexplore.ieee.org/Xplore/login.jsp?url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F5492928%2F5494277%2F05494315.pdf%3Farnumber%3D5494315&authDecision=-203>. Acesso em: 12 dez. 2010.

[Jaeger et al. 2010] JAEGER, Trent; SCHIFFMAN, Joshua. Outlook: Cloudy with a Chance of Security Challenges and Improvements, *IEEE Security and Privacy*, v. 8, n. 1, p. 77-80, jan./fev. 2010. Disponível em: <http://doi.ieeecomputersociety.org/10.1109/MSP.2010.45>. Acesso em: 26 dez. 2010.

[Kaushik 2009] KAUSHIK, Nishant. IdM And The Cloud: Stormy Days Ahead. In: *Oracle OpenWorld* 2009. Disponível em: <http://blog.talkingidentity.com/2009/10/screencast-of-my-openworld-session-on-idm-and-the-cloud.html>. Acesso em: 26 dez. 2010.

[Leaf 2010] LEAF, Dawn. *Overview*: NIST Cloud Computing Efforts. 2010. Disponível em: <http://csrc.nist.gov/groups/SNS/cloud-computing/documents/forumworkshop-may2010/nist_cloud_computing_forum-leaf.pdf>. Acesso em: 10 dez. 2010.

[Lee et al. 2009] LEE, Hyangjin; JEUN, Inkyoung; JUNG, Hyuncheol. Criteria for Evaluating the Privacy Protection Level of Identity Management Services. Emerging Security Information, Systems and Technologies, 2009. SECURWARE '09. In: Third International Conference on p. 155-160, p. 18-23. Jun. 2009. Disponível: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5211014&isnumber=5210974>. Acesso em: 26 dez. 2010.

[Marcon et al. 2010] MARCON JR.; ARLINDO L. Aspectos de Segurança e privacidade em ambientes de computação em nuvem. In: SIMPÓSIO BRASILEIRO EM SEGURANÇA DA INFORMAÇÃO E DE SISTEMAS COMPUTACIONAIS, 10., 2010, Fortaleza. *Anais eletrônicos*... Fortaleza: SBSeg, 2010. Minicurso. Disponível em: <http://www.insert.uece.br/sbseg2010/anais/04_minicursos/minicurso_02.pdf>. Acesso em: 01 dez. 2010.

[Mather et al. 2009] MATHER, Tim; KUMARASWAMY, Subra; LATIF, Shahed. *Cloud Security and Privacy:* An Enterprise Perspective on Risks and Compliance. O'Reilly Media. 2009.

[Mell; Grance 2009] MELL, P.; GRANCE, T. The NIST definition of cloud computing. National Institute of Standards and Technology. 2009.

[Mont 2010] MONT, Marco Casana. The Future of Identity in the Cloud: requirements, risks & opportunities. In: EEMA E-IDENTITY CONFERENCE, 2009, *Anais eletrônicos*... Disponível em: <http://www.hpl.hp.com/personal/Marco_Casassa_Mont/Documents /Papers/HPL-IdentityCloud%20-%20marcocasassamont.ppt>. Acesso em: 06 maio 2010.

[Moreira et al. 2010] MOREIRA, E. Q. et al. *Federação CAFe*: implantação do provedor de identidade. Brasilia: Escola Superior de redes RNP, 2010.

[Nurmi et al. 2009] NURMI, D. et al. The eucalyptus open-source cloud computing system. Em IEEE/ACM International Symposium on Cluster Computing and the Grid. 2009.

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

[OASIS 2005b] OASIS. EXtensible Access Control Markup Language (XACML) version 2.0. *Organization for the Advancement of Structured Information Standards (OASIS)*. 2005b. Disponível em: <http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf>. Acesso em: 26 dez. 2010.

[OASIS 2005c] OASIS. Security Assertion Markup Language (SAML) 2.0. Technical *Overview*. 2005c.

[OASIS 2007] SECURITY ASSERTION MARKUP LANGUAGE –SAML. V2.0. Oasis, 2007. Disponível em: <http://docs.oasis-open.org/security/saml/v2.0/saml-conformance-2.0-os.pdf>. Acesso em: 11 mar. 2010

[OpenAM 2010] OPENAM 2010. In: FORGEROCK. Disponível em: <http://www.forgerock.com/openam.html>. Acesso em: 15 dez. 2010

[OpenID 2010] OPENID. *OpenID allows you to use an existing account to sign in to multiple websites, without needing to create new passwords*. Disponível em: <http://openid.net/get-an-openid/what-is-openid/>. Acesso em: 20 dez. 2010

[Pearson 2009] PEARSON, Siani. Taking account of privacy when designing cloud computing services. In: WORKSHOP ON SOFTWARE ENGINEERING CHALLENGES OF CLOUD COMPUTING. IEEE Computer Society, Washington, DC, USA. 2009.

[Pearson et al. 2009] PEARSON, S.; SHEN, Y.; MOWBRAY, M. A privacy manager for cloud computing. *Cloud Computing*, v. 5931, 2009. p. 90–106.

[Zhang et al. 2010] ZHANG, Q.; CHENG, L.; BOUTABA, R. Cloud computing: state-of-the-art and research challenges. *Springer Journal of Internet Services and Applications*. 2010.

[Zhang, Zhou 2009] ZHANG, L.; ZHOU, Q. CCOA: Cloud computing open architecture. In: INTERNATIONAL CONFERENCE ON WEB SERVICES. 2009.