UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE ELETRICIDADE

Luiz Aurélio Batista Neto

Um Mecanismo de Integração de Identidades Federadas entre Shibboleth e SimpleSAMLphp para aplicações de Nuvens

São Luís
2014

Luiz Aurélio Batista Neto

Um Mecanismo de Integração de Identidades Federadas entre Shibboleth e SimpleSAMLphp para aplicações de Nuvens

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão como requisito parcial para a obtenção do grau de MESTRE em Engenharia de Eletricidade.

Orientador: Zair Abdelouahab

Doutor em Ciências da Computação – UFMA

São Luís

2014

Batista Neto, Luiz Aurélio

Um Mecanismo de Integração de Identidades Federadas entre Shibboleth e SimpleSAMLphp para aplicações de Nuvens / Luiz Aurélio Batista Neto. – São Luís, 2014.

106 f.

Orientador: Zair Abdelouahab.

Impresso por computador (fotocópia).

Dissertação (Mestrado) – Universidade Federal do Maranhão, Programa de Pós-Graduação em Engenharia de Eletricidade. São Luís, 2014.

1. Autenticação - Segurança. 2. Computação em Nuvem. 3. Identidades Federadas. I. Título.

CDU 004.056.23

UM MECANISMO DE INTEGRAÇÃO DE IDENTIDADES FEDERADAS ENTRE SHIBBOLETH E SIMPLESAMLPHP PARA APLICAÇÕES DE NUVENS

Luiz Aurélio Batista Neto

Dissertação aprovada em 19 de setembro de 2014.

Prof. Zair Abdelouahab, Ph.D.
(Orientador)

Prof. Rafael Fernandes Lopes, Dr.
(Membro da Banca Examinadora)

Prof. Aristófanes Corrêa Silva, Dr.
(Membro da Banca Examinadora)

A minha mãe Concebida pelo amor incondicional.

Resumo

Aplicações de Computação em Nuvem estão vulneráveis a ameaças de segurança oriundas da Internet, por conta do compartilhamento de recursos com outros usuários e gerenciados por terceiros. A diversidade de serviços e tecnologias se apresenta ainda como desafio para integração de identidades e dados de usuários no contexto distribuído. Para lidar com essas questões, técnicas de gerenciamento de identidades, especialmente as que utilizam a abordagem federada, se mostram fundamentais para proteger as informações de acessos não autorizados e permitir o intercâmbio de recursos entre as diferentes partes confiáveis entre si. O objetivo deste trabalho é desenvolver um modelo que permita a integração entre provedores de identidades por meio do protocolo *Security Assertion Markup Language (SAML)*, com a finalidade de prover o acesso a aplicações em múltiplos domínios de Computação em Nuvem. Neste cenário, cada domínio agrupa usuários e serviços conforme o mecanismo de representação do usuário de acordo com o sistema de gerenciamento de identidades utilizado (*Shibboleth* ou *SimpleSAMLphp*). O modelo proposto é implementado para verificar a sua aplicabilidade. Nos experimentos realizados por simulação computacional, os resultados obtidos demonstram a viabilidade do modelo apresentado.

Palavras-chaves: Segurança Computacional, Computação em Nuvem, Identidades Federadas

Abstract

Cloud computing applications are vulnerable to security threats originating from the Internet, because of the resources with other users and managed by third parties sharing. The diversity of services and technologies still presents a challenge to identity integration and user data in the distributed context. To address these issues, identity management techniques, especially those using a federated approach, appear crucial to protect the information from unauthorized access and allow the exchange of resources between the different trusted parties among themselves. The objective of this work is to develop a model that allows integration between identity providers through the Security Assertion Markup Language (SAML) protocol, in order to provide access to applications in multiple domains of Cloud Computing. In this scenario, each domain users and groups services as the mechanism of representation of the user according to the identity management system used (Shibboleth or SimpleSAMLphp). The proposed model is implemented to verify its applicability. In the experiments by computer simulation, the results obtained demonstrate the feasibility of the presented model.

Keywords: Security Computer, Cloud Computing, Federated Identity

Agradecimentos

As primeiras palavras de agradecimento vão a Deus, por manifestar Sua glória fornecendo-me condições para superar os desafios encontrados durante minha trajetória de vida e agora me permitindo alcançar mais este sonho.

A meu orientador Prof. Ph.D. Zair Abdelouahab, um brilhante ser humano e educador, pela oportunidade concedida, por seu apoio competente e caloroso, pelos enriquecedores conselhos e troca de ideias com que me presenteou ao longo deste trabalho. A este, exprimo a viva e cordial gratidão.

A Profa. Dra. Christiane Lima, pelo fundamental auxílio na co-orientação desta pesquisa, por sua paciência, dedicação, experiência partilhada, e especialmente pela revisão e correção desta dissertação; dirijo-lhe minha grata admiração.

A minha família que sempre me deu suporte e confiança. Obrigado, mamãe (Concebida), papai (Luiz Carlos), irmãos (Duduca e Priscilla), outra mãezona (Dadá), ao amigão (Roberto Costa) e sua família, por toda dedicação, carinho e apoio dispensados a mim de inúmeras formas: orações, palavras, gestos e ações, essenciais à minha existência.

À Mayana Santana e a sua família, pelo carinho e suporte em muitos momentos dessa jornada. A meu irmãozão Josivaldo, que se dispôs a ser meu motorista particular durante os 3 primeiros meses de mestrado, quando me encontrava em início de recuperação após cirurgia devido a uma fratura no tornozelo. A estas pessoas, minha valorosa gratidão.

Aos queridos colegas do Laboratório de Sistemas em Arquiteturas Computacionais (LABSAC) da UFMA, pelo essencial apoio a esta pesquisa e pelas alegrias partilhadas. Obrigado, Dhully, Marcos Sá, Willian "Cloudman", Higo "Shiboman", Jhonatan "Yuri", Mário "Locão", Leonardo "Capela", Jean "Pablito", Steve, Bruno, Cláudio, Renato, Gleison e Wagner. E ainda aos colegas de mestrado, Marcus Vinicius, Arikleyton, Felipe Maia, Berto, Suzane, Fhillipi e Alcides, pela parceria amistosa durante os trabalhos.

Ao Programa de Pós-Graduação em Engenharia de Eletricidade, pela oportunidade da realização do curso; a Profa. Dra. Maria da Guia, pelo diálogo científico proveitoso, e a Alcides e Neto, pelo bom atendimento às solicitações acadêmicas.

A REUNI (Reestruturação e Expansão das Universidades Federais), pelo financiamento da bolsa de estudo. A CAPES, exprimo meu reconhecimento.

A todos meus professores do IFMA. Especialmente, a Dra. Eveline Sá, Dr. Omar Andres Carmona Cortes e Dr. João Carlos, que forneceram minhas cartas de recomendação exigidas por este Programa de Pós-Graduação. A estes mestres, com carinho, exprimo meu respeito, admiração e gratidão.

Ao SENAI-MA, pela licença a estudo concedida. Em especial, à Valéria Carneiro, Celso de Oliveira e Raimundo Ribeiro. E ainda à Faculdade Pitágoras, especialmente a Deyvid Massoli (Coordenador do Curso de Redes) e Luzinete Menezes, pelo incentivo e liberação. A todos estes, minha apreciável gratidão.

Enfim, agradeço a meus familiares, amigos, demais professores e irmãos das Igrejas Batistas "Tabernáculo" e de "Pindaré-Mirim", que contribuíram com mensagens de incentivo e orações. E a todos aqueles que direta ou indiretamente contribuíram para a realização deste trabalho.

"Qualquer pessoa pode elevar-se acima das suas circunstâncias e alcançar sucesso, caso se dedique e tenha entusiasmo pelo que faz."

Nelson Mandela

Lista de Figuras

Lista de Tabelas

Lista de Siglas

XCP	Xen Cloud Plataform.
CSA	Cloud Security Alliance.
ENISA	European Network and Information Security Agency.
IaaS	Infraestrutura como Serviço.
IDC	International Data Corporation.
IdP	Identity Provider.
NIST	National Institute of Standards and Technology.
PaaS	Platforma como Serviço.
SaaS	Software como Serviço.
SAML	Security Assertion Markup Language.
SP	Service Provider.
SSO	Single Sign On.
XACML	eXtensible Access Control Markup Language.

Sumário

1 Introdução

Neste capítulo é apresentada uma descrição desta dissertação, com a finalidade de fornecer uma visão geral dos problemas tratados e dos objetivos principais do trabalho de pesquisa realizado. Em seguida, é apresentada a estrutura do documento, com uma descrição resumida do conteúdo abordado em cada capítulo.

1.1 Contexto e Justificativa

O fenômeno da computação em nuvem é considerado o ingrediente principal de maior transformação da indústria de TI da atualidade. De acordo com o International Data Corporation (IDC) [39], os gastos mundiais com serviços de computação em nuvem atingiram 47,4 bilhões dólares em 2013 e as expectativas são que ultrapassem o valor de 107 bilhões de dólares em 2017.

Uma das principais vantagens da computação em nuvem é a possibilidade de pagamento somente pelos recursos contratados pelo usuário, uma ideia que foi prevista por John McCarthy em 1961 [51], "*computação pode um dia ser disponível como uma utilidade pública*". Esta característica resultou em outra: a capacidade de utilização de serviços sem a necessidade de instalação e manutenção por parte do usuário.

Em geral, autores como [54, 77, 98] definem a computação em nuvem como um modelo em que os usuários têm acesso sob demanda a um conjunto de recursos computacionais compartilhados e configuráveis, tais como redes, armazenamento, serviços ou aplicações que podem ser rapidamente provisionados e liberados com um mínimo de esforço de gerenciamento, utilizando a Web por meio de qualquer dispositivo (celulares, *tablets*, *notebooks*, etc).

Hoje, há diversos provedores que entregam serviços de computação em nuvem para usuários: *Amazon Web Services, Microsoft Azure, Google Apps, IBM*, entre outros. Também para desenvolvedores e pesquisadores, existem *sofwares open-source*, tais como: *Eucalyptus, OpenNebula, Nimbus e Xen Cloud Plataform (**XCP**)*.

Um provedor de computação em nuvem pode utilizar um ou mais modelos para a oferta de serviços tais como Infraestrutura como Serviço (IaaS), Platforma como Serviço (PaaS) ou Software como Serviço (SaaS). No IaaS, o consumidor contrata uma infraestrutura de hardware podendo escolher a capacidade de configuração (ex.CPU, memória, discos), sistema operacional (ex.Linux, Windows) e quaisquer *softwares* - tipo de serviço oferecido, por exemplo, pela Amazon [3]. No PaaS, o consumidor contrata uma plataforma com um conjunto de ferramentas específicas, por exemplo linguagens de programação como um tipo de serviço oferecido pela *Google Apps* [31]. E no SaaS, o consumidor contrata a utilização de uma aplicação hospedada num ambiente de computação em nuvem, como o *Gmail* [30].

Neste contexto, são necessárias funcionalidades para gerenciar o fluxo de identidade do usuário (ou seja, autenticação e autorização) e para controlar o acesso de usuários aos serviços consumidos por ele dentro do ambiente de computação em nuvem. Esses serviços estão suscetíveis a combinar dados de modelos de serviços de Computação em Nuvem (IaaS, PaaS e SaaS), pertencentes a diferentes domínios, em que cada um tem seu próprio mecanismo de controle de acesso. Por essa razão, o gerenciamento de identidades deve ser posto em prática para implantação de computação em nuvem, cuja interação deve ocorrer de forma transparente e segura.

Especialmente, visando controlar o acesso de usuários em diferentes domínios, o gerenciamento de identidades federadas (*Federated Identifier Management - FIM*) é considerado por pesquisadores e especialistas como um importante ativador de segurança, uma vez que irá desempenhar um papel vital para permitir a escalabilidade global que é necessária para a implantação com sucesso da computação em nuvem. Este tópico tem sido aplicado em vários ambientes, assim como os Recursos *Web*, *Webservices*, e Grades Computacionais, bem como tem sido investigado por vários órgãos de padronização como *Cloud Security Alliance (CSA), European Network and Information Security Agency (ENISA), National Institute of Standards and Technology (NIST)* de padronização envolvidos na computação em nuvem [73].

O gerenciamento de identidades federadas tem que garantir que as identidades acessem o serviço disponível de maneira transparente. Outro aspecto importante que está se tornando cada vez mais urgente, especialmente nos últimos tempos, são os casos de roubo de identidade e acesso indevido, que têm aumentado de forma constante nos últimos anos.

Por essa razão, deve ser tarefa de um bom sistema de gerenciamento de identidades poder tratar estes problemas para assegurar um elevado nível de segurança e privacidade. Além disso, a usabilidade é certamente um dos grandes desafios que precisam ser tratados nesse tipo de sistema, isto é, usuários devem acessar aplicações em domínios federados por meio de uma única autenticação, facilitando gerência de acesso tanto do ponto de vista do usuário quanto do provedores de identidades.

1.2 Motivação e Problemática

Empresas tem interesse em mudar sua infraestrutura para modelos de computação em nuvem (IaaS, PaaS, SaaS). Entretanto, elas ainda se preocupam com os riscos de segurança implícitos à incorporação de seus recursos dentro deste ambiente. A segurança em computação em nuvem desperta forte interesse na indústria e na área acadêmica. Evidências disso podem ser vistas nos trabalhos recentes de [47,71,94–96], que exploram especialmente a segurança como aspecto essencial para utilização dessa tecnologia.

Para muitos, a computação em nuvem começou em 2008, com a rápida disseminação no uso de *Webservices* e armazenamento da *Amazon*. Desde então, estudos vem sendo realizados no intuito identificar o estado da computação em nuvem, levantando desafios a serem enfrentados.

Uma pesquisa do *IDC* [40] com 244 executivos de TI investigou qual o aspecto mais preocupante quanto ao uso de serviços de computação em nuvens. Os resultados mostraram que 87,5% das pessoas entrevistadas preocupam-se com a segurança, apontando esta como uma das principais barreiras para adoção desse paradigma. A Figura 1.1 mostra os resultados dessa pesquisa.

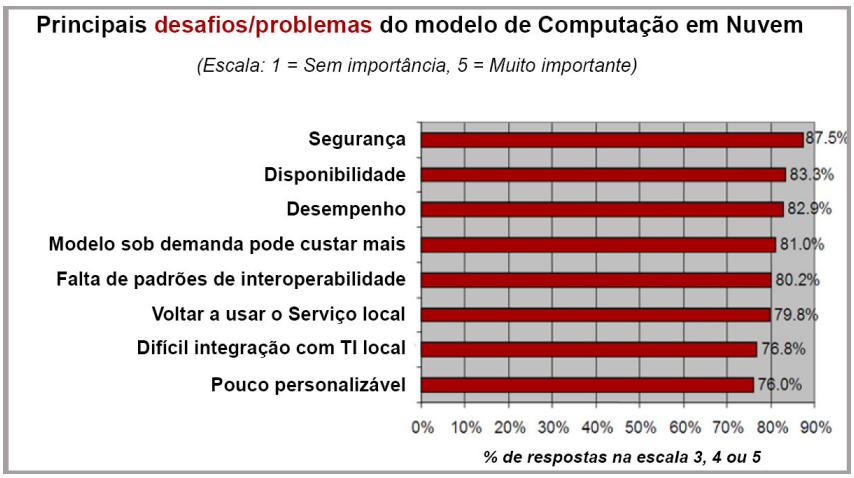

Figura 1.1: Preocupações inerentes à computação em nuvem. Adaptado de [40].

Em 2013, a *CSA* publicou o documento "*Cloud Computing Vulnerability Incidents: A Statistical Overview*" [68]. Na tentativa de verificar confiabilidade do ambiente de computação em nuvem, foram revisados 11.491 artigos de notícias sobre computação em nuvem em 39 fontes de janeiro de 2008 a fevereiro de 2012), relatando vulnerabilidades na área da computação em nuvem.

Durante esse período, o número de incidentes de vulnerabilidade no ambiente de computação em nuvem aumentou consideravelmente (conforme Figura 1.2). Por exemplo, esse número aumentou mais que o dobro ao longo de um período de quatro anos, passando de 33 em 2009 para 71 em 2011. Um total de 172 incidentes de vulnerabilidade no ambiente de computação em nuvem foram descobertos, dos quais 129 (75%) declararam suas causas, enquanto 43 (25%) não o fizeram.

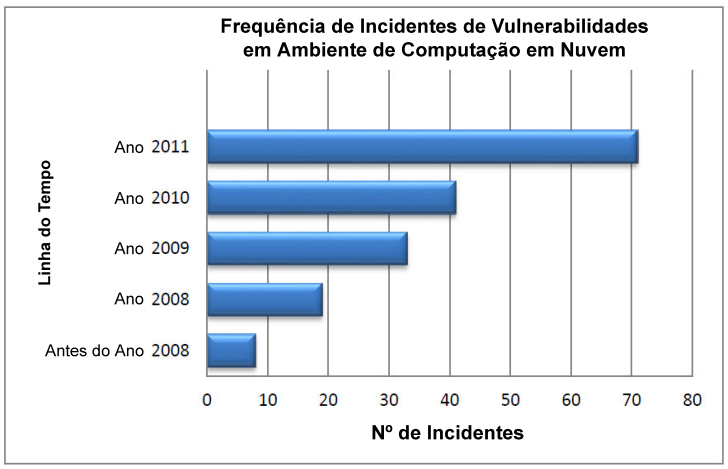

Figura 1.2: Frequência de incidentes em Computação em Nuvem. Adaptado de [18].

A pesquisa revelou três principais ameaças: “Interfaces e APIs inseguras” (51 incidentes; 29% de todas ameaças)”, “perda de dados” (43 ocorrências, 25%), e “falha de hardware” (18 ocorrências , 10%). Estas três ameaças foram responsáveis por 64% de todos os incidentes de vulnerabilidades no ambiente de computação em nuvem.

Os principais elementos para a segurança da computação em nuvem são mostrados [51] (Identidades, Infraestrutura e Informação) a seguir na Figura 1.3.

Figura 1.3: Principais elementos para segurança em computação em nuvem. Adaptado de [51].

O primeiro elemento da Figura 1.3 (segurança da identidade do usuário e dos serviços de computação em nuvem) é o propósito desta pesquisa. Este trabalho foca no gerenciamento de identidades federadas, como categoria do campo de estudo da segurança em computação em nuvem.

De acordo com [33, 84], a segurança computacional se apresenta como um requisito indispensável para garantir o sucesso em ambientes de Computação em Nuvem, e [65] destaca a proteção à privacidade, já que dados sensíveis passam a ficar sob a custódia de terceiros. Nesse contexto, o gerenciamento de identidades cresce em importância conforme crescem os serviços que precisam utilizar autenticação e autorização para controlar o acesso de usuários [5, 8].

Aplicações de computação em nuvem são entregues a usuários por um grande número de provedores de serviços e utilizando diferentes tipos de tecnologias, abstraindo um cenário heterogêneo em nível de *hardware* ou *software* [96]. Esse cenário

inspirou a concentração desta pesquisa na área de segurança em computação em nuvem, dentro da qual foram identificados dois desafios relativos ao Gerenciamento de Identidades Federadas [21] [24].

O **primeiro desafio** refere-se a um dos aspectos mais críticos da segurança em computação em nuvem, que é a privacidade dos dados do usuário que acessa as aplicações [96]. Provedores de computação em nuvem dispõem de múltiplos serviços, tais como *e-mails*, desenvolvimento de aplicativos personalizados, armazenamento de dados e gestão de infraestrutura, que podem se caracterizar como sendo um alvo propício a ataques por potenciais invasores. Ameaças como esta podem afetar diretamente as propriedades da segurança computacional, a saber: disponibilidade, confidencialidade e integridade. Dessa forma, essas ameaças pode vir a comprometer todo o ambiente de computação em nuvem [70].

Vários autores como [23, 29, 44, 65, 93] destacam que a privacidade precisa ser tratada no contexto de computação em nuvem. De acordo com as instituições de pesquisa CSA, ENISA e o NIST, um dos aspectos fundamentais de segurança é o uso de Interfaces de acesso às nuvens, das quais se destacam: as *Interfaces de Usuários*, como meios de utilização de serviços; e as *Interfaces de Autenticação*, responsáveis pela execução de tarefas administrativas e de controle de acesso aos sistemas.

O **segundo desafio** é a falta de interoperabilidade entre diferentes Sistemas de Gerenciamento de Identidades. Validar um único usuário nestas ferramentas de maneira mútua impõe barreiras de integração que vão desde protocolos, linguagens de programação, plataformas (Sistema Operacional, provedor de computação em nuvem e virtualização) usadas no projeto de TI, e até as diversas infraestruturas de segurança adotadas.

Na literatura científica, são encontrados alguns trabalhos que abordam essa problemática. Segundo a *CSA*, a falta de interoperabilidade de gerenciamento de identidades em computação em nuvem é gerada pelo fato de cada ferramenta para gerência de recursos utiliza uma solução própria para gerir as identidades e o acesso a recursos. O trabalho de [21] justifica que este problema se refere a diferentes provedores de serviços utilizarem diferentes mecanismos de mapeamento de usuários em contextos de autenticação. E, por fim, [32] [96] citam a dificuldade de migração dos recursos ou serviços de um ambiente de computação em nuvem para outro.

Em virtude disso, dado o contexto de federação/cooperação ao qual os provedores de computação em nuvem estão situados, o uso de padrões é um requisito vital para o funcionamento adequado dessas soluções [24]. Por exemplo, para a realização de autenticação única (*Single Sign On*) entre provedores de computação em nuvem, a *CSA* recomenda que os provedores de serviços sejam flexíveis e que aceitem os formatos utilizados pelos provedores de identidades. Por outro lado, [24] propõe a utilização de padrões de gerenciamento de identidades, e aponta, como exemplo, soluções de conectividade da *Microsoft* e os padrões abertos *OAuth*, *OpenID* e *SAMLv2*, entre outros.

1.3 Hipótese e Objetivos

Este trabalho consiste em prover a integração de identidades federadas tendo como premissa fundamental a melhoria da segurança da informação sobre o acesso dos usuários às aplicações contidas em ambientes de Computação em Nuvem, uma vez que os mecanismos de controle de acesso são usados para garantir a privacidade dos dados dos usuários e usabilidade por meio do processo *Single Sign On (SSO)* ou "Autenticação única". Desse modo, a principal hipótese desta pesquisa pode ser definida como segue.

Considerando a importância da interoperabilidade de sistemas computacionais em ambientes distribuídos, é possível utilizar tecnologias, protocolos e padrões de gerenciamento de identidades de forma a torná-los mais integráveis e aplicar tais sistemas em ambientes de Computação em Nuvem.

Sendo assim, esta pesquisa considera a necessidade da elaboração de um modelo com a finalidade de suportar a interoperabilidade entre distintos provedores de identidade; e ser aplicável em um ambiente de Computação em Nuvem. O modelo proposto nesta pesquisa tem como objetivo viabilizar o acesso seguro do usuário a aplicações da camada SaaS localizadas em domínios distintos, cuja infraestrutura é baseada em plataformas de nuvem.

Para efetuar a validação da hipótese desta dissertação, foi traçado um conjunto de objetivos, os quais são descritos a seguir.

O **objetivo geral** deste trabalho é desenvolver um modelo que permita a integração de identidades federadas entre diferentes provedores de identidade (IdPs) para acesso a aplicações de Computação em Nuvens por meio de uma única autenticação (*Single Sign On*).

A exploração das características e princípios de gerenciamento de identidades e do paradigma de Computação em Nuvem possibilitou a construção de um modelo de segurança computacional capaz de prover a interoperabilidade entre os provedores de identidade para acesso às aplicações em múltiplas nuvens. Essa interoperabilidade ocorre entre diferentes IdPs pelo uso do protocolo *Security Assertion Markup Language (SAML)*, de modo que a comunicação entre os componentes do modelo proposto e os Sistemas de Gerenciamento de Identidades Federadas utilizados ocorra de forma segura e transparente a usuários e provedores de serviços.

Nesse modelo, cada domínio inclui usuários e provedores de serviços de acordo com o sistema de gerenciamento de identidades adotado. No desenvolvimento do modelo, tem-se como pressupostos fundamentais o protocolo *SAML*, modelos de gerenciamento de identidades federadas e as tecnologias de Computação em Nuvem, como exemplo, a virtualização e os Serviços *Web*.

A partir das premissas citadas, esta pesquisa foi norteada pelos seguintes **objetivos específicos**:

- Projetar um modelo que viabilize a integração de identidades federadas para realização do *SSO* entre diferentes IdPs em ambiente de múltiplas nuvens;
- Apresentar um fluxo de autenticação federada em nuvens;
- Implementar um protótipo baseado no modelo proposto;
- Comprovar a aplicabilidade desse modelo em ambiente de Computação em Nuvem.

1.4 Estrutura da Dissertação

A redação deste trabalho está estruturada cinco capítulos. Este capítulo descreveu o contexto geral do trabalho, a motivação e os objetivos. Os demais capítulos estão dispostos da seguinte forma:

- **Capítulo 2:** está organizado em 4 seções. A primeira seção *Segurança da Informação* apresenta os principais conceitos de segurança da informação, técnicas de segurança existentes e os princípios de controle de acesso. A segunda seção *Computação em Nuvem* descreve uma visão geral desse paradigma, destacando características, modelos e seus aspectos de segurança. A terceira seção descreve os conceitos e abordagens gerais do *Gerenciamento de Identidades*, especialmente fundamentais para o desenvolvimento deste trabalho. E, por fim, os *Trabalhos Relacionados* que serviram de base para esta dissertação;

- **Capítulo 3:** descreve o modelo desenvolvido, mostrando os diagramas especificados e o funcionamento esperado de cada um. Também é apresentada uma análise comparativa dos trabalhos relacionados com o modelo proposto por esta pesquisa;

- **Capítulo 4:** descreve os aspectos de implementação do modelo proposto, mostrando cenário de testes e como cada elemento implementado está relacionado com os diagramas especificados no modelo proposto. Também são apresentados os testes e os resultados obtidos;

- **Capítulo 5:** realiza uma conclusão deste trabalho, por fim, ressaltando suas contribuições e fazendo uma projeção de pesquisas futuras.

2 Fundamentação Teórica

Este capítulo elenca os conhecimentos fundamentais para o entendimento desta dissertação. A primeira seção aborda os principais conceitos de Segurança da Informação; a segunda seção descreve o paradigma da Computação em Nuvem, fornecendo conceitos, modelos e perspectivas de segurança. A terceira seção apresenta conceitos e abordagens gerais de Gerenciamento de Identidades, essenciais para o desenvolvimento deste trabalho. E, por fim, a última seção apresenta Trabalhos Relacionados. Tais pressupostos serviram de base para o desenvolvimento do modelo proposto desta pesquisa.

2.1 Segurança da Informação

A garantia da segurança da informação em ambientes distribuídos como a *Internet* constitui, cada vez mais, uma preocupação mundial, englobando desde os organismos públicos, privados, universidades, empresas até os cidadãos, de forma individual ou coletiva. Seja em bibliotecas ou arquivos digitais, na banca eletrônica, no *e-learning* (ensino à distância) ou em qualquer outra área, a segurança da informação é uma questão chave para a sobrevivência de muitas organizações.

Nesse cenário, computadores e dispositivos portáteis em diversos lugares do mundo têm acesso às informações por meio de avançadas tecnologias e da *Internet*, aumentando a complexidade e importância de proteção das informações. O nome genérico para o conjunto de ferramentas projetadas para proteger dados e impedir *hackers* é a segurança da informação [85].

Autores geralmente definem a *segurança da informação* baseada em suas propriedades (confidencialidade, integridade e disponibilidade) ou em defesa contra ameaças que impõem riscos aos negócios da organização [6, 97]. A seguir são apresentadas duas definições para o termo segurança da informação:

- É a prática de assegurar que os recursos que geram, armazenam ou proliferam as informações, sejam protegidos contra a quebra de confidencialidade,

comprometimento da integridade e contra a indisponibilidade de acesso a tais recursos [97];

- É a proteção da informação de vários tipos de ameaças para garantir a continuidade do negócio, minimizar o risco ao negócio, maximizar o retorno sobre os investimentos e as oportunidades de negócio [6].

A IBM desenvolveu um *framework* que apresenta os níveis de segurança das arquiteturas de TI que devem ser estabelecidos, conforme ilustrados na Figura 2.1.

Figura 2.1: Níveis segurança da arquitetura de TI. Adaptado de IBM [37].

Segundo [55, 57, 92], a segurança da informação pode ser classificada como *segurança física* e *segurança lógica*, cujas explicações são fornecidas a seguir:

- **Segurança física:** em geral, refere-se à gestão de pessoas, equipamentos e instalações. Em relação à gestão de pessoas, estão as situações em que a componente humana constitui a principal fonte de atenção e de risco em situações como o erro, a falha humana, a fraude ou o roubo. Quanto aos equipamentos, referem-se ao *hardware* computacional (por exemplo, computadores, servidores, infraestruturas de rede) e outros equipamentos, tais como o de fornecimento de energia, sistemas de controle de acessos físicos, sistemas de detecção e combate a incêndios. Por fim, referente às instalações,

trata-se da perspectiva de engenharia e arquitetura, nomeadamente o local onde está fisicamente instalado o sistema que gere e armazena a informação digital. Refere-se ao nível *infraestrutura física* da Figura 2.1;

- **Segurança lógica:** semelhante à segurança física, também a segurança lógica é essencial para garantir a segurança da informação. Ainda que aparentemente menos visível, a segurança lógica deve estar em permanente atualização, de forma a acompanhar também a evolução dos riscos e das possíveis ameaças. Nesse aspecto, o princípio básico é que a segurança lógica deve ser implementada em cada nível da arquitetura de TI. Refere-se aos demais níveis da Figura 2.1, que são descritos a seguir.

  - **Pessoas e identidades:** as organizações precisam garantir que usuários autorizados por todas as empresas e cadeias de suprimentos tenham acesso aos dados e ferramentas necessárias, no momento que precisam, enquanto bloqueiam aqueles que não possuem autorização para acesso. O monitoramento de usuário com privilégios, que inclui as atividades de registro, se tornou uma importante necessidade. Ferramentas de gerenciamento de identidades são fundamentais para controlar a autenticação e autorização de acesso dos usuários;
  - **Dados e informações:** a proteção de dados é questão de segurança mais importante para muitas organizações. As preocupações típicas incluem a maneira com que os dados são armazenados e acessados, os requisitos de conformidade e auditoria, as questões de negócios que envolvem o custo das violações de dados, os requisitos de notificação e os danos ao valor da marca;
  - **Aplicativos e processos:** todos os requisitos de segurança (confidencialidade, integridade e disponibilidade) servem para aqueles aplicativos que se encontram em ambientes distribuídos como a rede. Isso inclui a garantia de que os serviços da Web publicados na Internet, por exemplo, sejam compatíveis com os níveis de segurança exigidos e atendam às políticas de negócios;
  - **Rede, Servidor e Terminal:** em ambientes distribuídos e compartilhados, é imprescindível garantir que todos os domínios estejam devidamente

isolados e que não exista a possibilidade de dados ou transações vazarem de um domínio para o outro. Conforme os dados saem do controle dos usuários, esses esperam que o ambiente possua recursos como detecção de invasão e sistemas de prevenção.

2.1.1 Terminologias

Vários termos citados em [6,48,52,53] aplicam-se à segurança da informação e tais significados são essenciais para compreensão desta seção. Portanto, são apresentados a seguir:

- **Ativo:** hardware, software ou qualquer patrimônio que tenha valor para a organização. Exemplos: dados, informações e equipamentos;
- **Usuários:** funcionários, prestadores de serviços, clientes, fornecedores, bolsistas e estagiários;
- **Vulnerabilidades:** são fragilidades de um ativo (componente humano, tecnológico, físico ou lógico) ou grupo de ativos que pode ser explorada por uma ou mais ameaças. Exemplos: contas sem senha, falhas em programas aplicativos ou sistemas operacionais (*buffer overflow* em serviços *DNS, SMTP,* etc.), injeção de comandos *SQL*;
- **Risco:** é a medida do custo de uma vulnerabilidade que incorpora a probabilidade de uma violação ocorrer. A análise do risco serve de base para a criação das políticas de segurança. Essa análise visa identificar quais os elementos que apresentam risco de falha/desaparecimento ou de falha parcial ou perda de desempenho, e ainda para quantificar o impacto de um ataque por meio de uma ameaça existente;
- **Tratamento de Riscos:** processo de seleção e implementação de medidas para modificar um risco;
- **Ameaças:** ações que podem afetar a segurança de bens corporativos e causar a destruição, divulgação, modificação de dados e/ou impedir o funcionamento de um sistema. Exemplo: hackers, crackers, agentes naturais;

- **Ataque:** representa a ação tomada por um agente capaz de explorar vulnerabilidades, causando riscos aos ativos da organização;

- **Atacantes:** definidos como agentes que visam o acesso não autorizado a recursos da organização. Esses agentes são ainda conhecidos como *hackers*, *crackers*, intrusos, invasores;

- **Gerador de ameaças:** são os atacantes propriamente ditos;

- **Incidentes de Segurança da Informação:** é indicado por um simples ou por uma série de eventos de segurança da informação indesejados ou inesperados, que tenham uma grande probabilidade de comprometer as operações do negócio e ameaçar a segurança da informação;

- **Política de Segurança:** é um conjunto de regras que especifica como um sistema fornece seus serviços, estabelece os limites de operação dos usuários do sistema e determina a maneira pela qual as informações e os recursos são administrados, protegidos e distribuídos. Pode ser vista por três aspectos [59]:

  - **Política de Segurança Física:** refere-se à situação física do sistema a proteger. Neste ramo, em particular, são definidas as medidas contra a violação de segurança física provocada por incêndio, catástrofes naturais etc;
  - **Política de Segurança Administrativa:** define os controles para o acesso lógico das informações, especificando "quem" tem acesso a "que" e em "quais" circunstâncias;
  - **Política de Segurança Lógica:** que pode ser decomposta em políticas de autenticação e políticas de autorização. Quando um usuário é identificado e autenticado, a política de autorização deve especificar quais são as operações que este usuário particular pode realizar no sistema.

Para alcançar este controle, é necessário ter clareza do fluxo de atividades que envolve termos-chave dentre os citados. A Figura 2.2 expressa a relação entre os principais termos compondo um modelo de fluxo de controle de segurança.

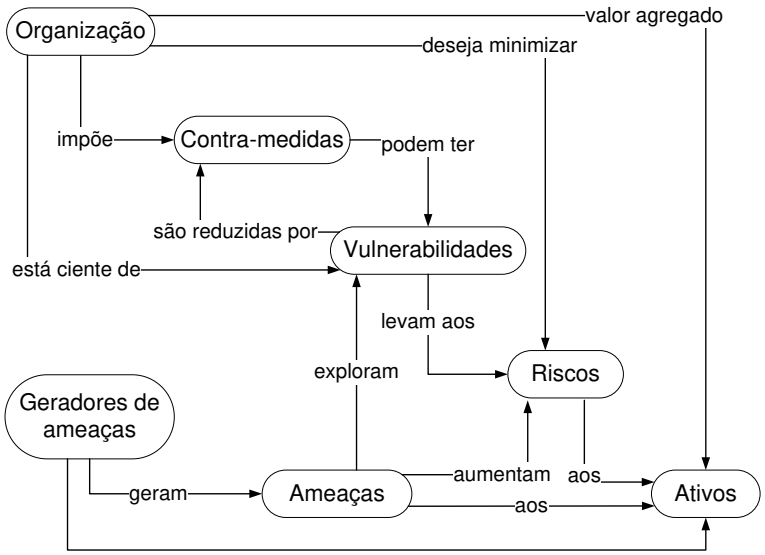

Figura 2.2: Fluxo de controle da segurança. Adaptado de [52].

Essa ilustração ainda pode ser usada pelas organizações como ponto de partida para o planejamento e desenvolvimento de suas políticas de segurança e regras de comportamento, que são descritas na Seção 2.1.4.

Por outro lado, tem-se ainda as propriedades da segurança como pontos-chave, que, num contexto mais amplo, servem de base para orientar a análise, o planejamento e a implementação de políticas de segurança para um determinado grupo de informações que se deseja proteger, e são essenciais para ter um projeto de segurança bem-sucedido, conforme Seção 2.1.2.

2.1.2 Propriedades

A segurança da informação, em geral, tem o objetivo de proteger os dados de usuários e organizações; tem o dever de estimular comportamentos seguros por meio do estabelecimento de regras e políticas de segurança; baseia-se na necessidade de se manter no sistema um conjunto de propriedades [52].

As três propriedades da segurança da informação são discutidas e referidas como “CID” (Confidencialidade, Integridade e Disponibilidade), conforme ilustra a Figura 2.3.

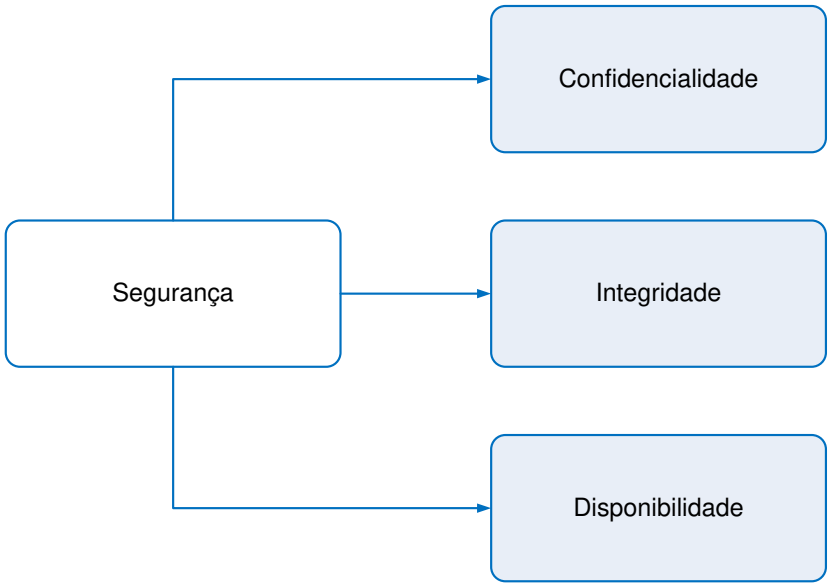

Figura 2.3: Propriedades da segurança. Adaptado de [92].

Cada propriedade tem sua função específica conforme as definições a seguir [53] [55]:

- **Confidencialidade ou privacidade:** propriedade necessária para garantir que a informação só será acessada por quem tem autorização. Um exemplo disto é quando um número de cartão de crédito vaza para outras fontes que não tinham autorização de ter aquele número.

Ataques de espionagem para capturar dados da rede é um caso comum de violação da confidencialidade da informação. Ferramentas que usam Criptografia, por exemplo, dão suporte à confidencialidade, pois são capazes de embaralhar ou codificar dados para impedir que usuários não autorizados os leiam ou adulterem. Assim, somente pessoas com acesso a uma senha ou chave podem descriptografar e utilizar os dados.

- **Integridade:** propriedade necessária para garantir que haverá consistência e completude dos dados quando acessados. Um exemplo de integridade é quando se transmite uma mensagem para alguma pessoa e no meio do caminho essa mensagem é adulterada e o conteúdo é modificado. Nesse caso houve o comprometimento da integridade da mensagem por uma fonte não autorizada;

Ataques de interceptação de mensagens podem culminar na modificação de dados e se configurar um caso de violação da integridade da informação. Ferramentas

criptográficas anteriormente mencionadas ajudam a para proteger a confidencialidade da informação, negando acesso de usuários a dados sem que tenham os direitos de acesso adequados, colaborando para evitar, antes de tudo, que dados sejam modificados.

Além disso, existem outras ferramentas especialmente projetadas para apoiar a integridade, incluindo as seguintes: Cópias de segurança, somas de verificação (função que mapeia o conteúdo de um arquivo para um valor numérico) e códigos de correção (métodos para armazenar dados de modo que pequenas alterações podem ser facilmente detectadas e automaticamente corrigidas);

- **Disponibilidade:** propriedade necessária para garantir que os usuários que têm autorização poderão acessar a informação quando necessário. Um exemplo de comprometimento da disponibilidade é quando se tenta fazer uma transação bancária e o sistema falha ou encontra-se indisponível.

Ataques de negação de Serviço ou *Denial of Service (DoS)* representam um caso comum de violação da disponibilidade do acesso à informação. Diversas ferramentas providenciam disponibilidade, incluindo as seguintes: proteções físicas (estruturas capazes de suportar tempestades, terremotos, explosões e enfrentar interrupções de energia) e redundâncias computacionais (computadores e dispositivos de armazenamento que servem como reserva no caso de falhas, por exemplo, *arrays* redundantes de discos (*RAID*)).

Além dos conceitos de Confidencialidade, Integridade e Disponibilidade, discutidos anteriormente, existem diversos conceitos adicionais importantes nas aplicações modernas de segurança de computadores. De forma semelhante, esses conceitos podem ser caracterizados por um acrônimo de três letras, "GAA", que se referem a Garantia, Autenticidade e Anonimato.

- **Garantia:** propriedade necessária que se refere como a confiança fornecida é gerenciada em sistemas de computação. Reconhecidamente, a própria confiança é difícil de quantificar, mas sabemos que ela envolve o grau de confiança em pessoas ou sistemas, quando se comportam segundo a política de segurança, isto é, da forma prevista;

- **Autenticidade:** propriedade necessária para verificação da identidade do cliente que solicita o uso dos dados. É ainda a habilidade de determinar que afirmações, políticas e permissões oriundas de pessoas ou sistemas são genuínas. Formalmente, diz-se que um protocolo que consegue autenticidade demonstra não repúdio.

  - **Não repúdio ou irretratabilidade:** propriedade necessária para que afirmações autênticas emitidas por alguma pessoa ou sistema não podem ser negadas pela autoria. A principal forma de efetivar esta propriedade é por meio do uso de assinaturas digitais (computações criptográficas que permitem a uma pessoa ou sistema se comprometer com a autenticidade de seus documentos de maneira única);

- **Anonimato:** propriedade necessária de que certos registros ou transações não sejam atribuíveis a qualquer indivíduo. Essa propriedade previne contra o perigo de espalhar identidades por meio de um hospedeiro de registros digitais, que associa a identidade a outros dados, como histórico médico, compras, registros legais, *e-mail*, etc. Se empresas precisam publicar dados sobre seus membros ou clientes, deve-se esperar que façam isso de modo a preservar a privacidade, usando uma das seguintes ferramentas:

  - **Agregação:** a combinação de dados de muitos indivíduos de modo que a divulgação dessas somas ou médias não possa ser vinculada a qualquer indivíduo. Exemplo: dados do IBGE não expõem detalhes sobre quaisquer indivíduos;
  - **Mistura:** o entrelaçamento de transações, informações ou comunicações de modo que não possam ser rastreadas a nenhum indivíduo. Exemplo: sistemas que podem misturar dados juntos de maneira quase aleatória, de forma que transações e pesquisas possam ser realizadas, mas sem a revelação de qualquer identidade individual;
  - **Representantes (*proxies*):** agentes de confiança que querem se engajar em ações para o indivíduo de maneira que não possam ser rastreados de volta para aquela pessoa. Exemplo: representantes de pesquisa na Internet são sites que fornecem a sua própria interface com o navegador, de modo

que indivíduos podem visitar sites que poderiam estar bloqueados, por exemplo, devido aos países onde estão localizados;

– **Pseudônimos:** identidades fictícias que podem servir como identidades reais em comunicações ou transações, mas que são conhecidas apenas por entidades de confiança. Por exemplo, muitos *sites* de redes sociais permitem que usuários interajam com os demais usando pseudônimos, de modo que eles podem se comunicar e criar um personagem *online* sem revelar sua verdadeira identidade;

Entretanto, é necessário compreender ainda os tipos de ameaça que podem comprometer essas propriedades.

2.1.3 Ameaças e Ataques

Os atacantes são a peça-chave das ameaças, e podem ser classificados como internos ou externos. Um atacante interno pode ser um usuário legítimo que consegue se passar por membros da rede, enquanto que os externos são aqueles que influenciam, mas não participam da rede. De fato, a eficiência e as possibilidades de ataques variam de acordo com o acesso que o atacante tem à rede [25].

Conforme visto na Seção 2.1.1, a violação da segurança é decorrente de ataques que exploram vulnerabilidades ou fragilidades dos sistemas agravando riscos às informações dentre outros ativos pertencentes a organizações ou indivíduos.

A Figura 2.4 ilustra as quatro categorias de ataques geralmente identificados em sistemas distribuídos, segundo [79]. Primeiramente, é considerado o fluxo normal de informação de um computador de origem se comunicando com um computador de destino (ilustrados pelos círculos), como se observa na Figura 2.4a. Cada uma das categorias de ataque é descrita na sequência.

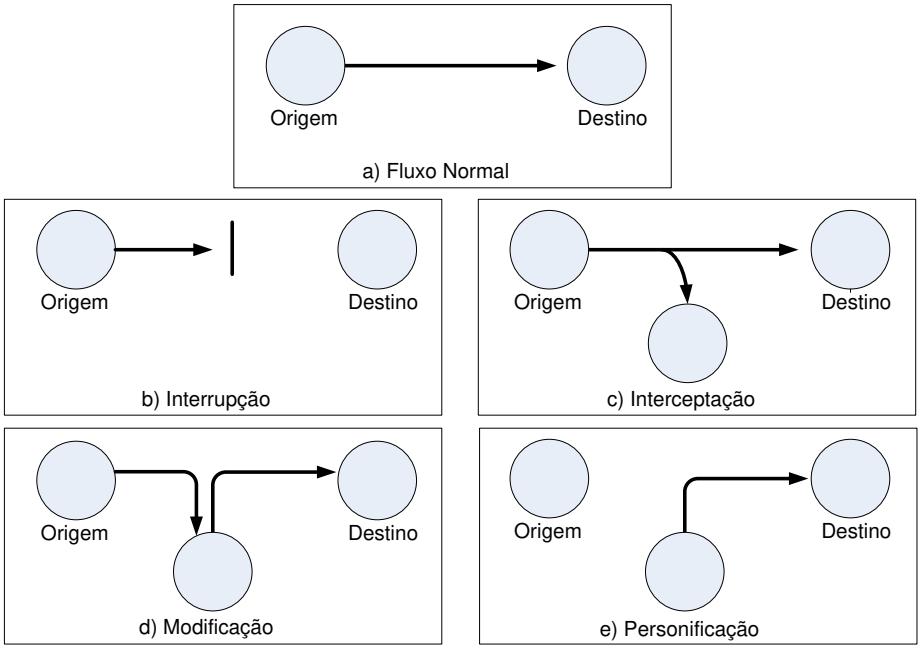

Figura 2.4: Categorias de Ataques [79].

- **Interrupção:** um componente ativo do sistema é destruído, ou torna-se indisponível ou inutilizável. Esse ataque afeta a propriedade da disponibilidade. Como exemplos têm-se a destruição de um equipamento de hardware, como um disco rígido ou o corte da linha de comunicação;

- **Interceptação:** uma entidade não autorizada ganha acessos aos componentes ativos, permitindo a captura de informações sigilosas como *usuário e senha* em uma rede. Compromete a confidencialidade, porque a informação é lida, indevidamente, por alguém;

- **Modificação:** uma entidade não autorizada não apenas obtém acesso, mas falsifica componentes ativos. Este é um ataque sobre a integridade. Exemplos são a mudança dos dados em um arquivo ou modificação na mensagem sendo transmitida em uma rede;

- **Personificação:** também conhecida como *fabricação*; uma terceira parte insere dados falsos no sistema, podendo se passar por uma parte confiável. Este é um ataque sobre a autenticidade.

Ataques à segurança podem ainda ser classificados como ataques *ativos* e *passivos*. Nos *ataques passivos*, o atacante não interfere no funcionamento da rede, mas pode escutá-la e analisar o seu tráfego. O atacante tem acesso à informação, porém não a altera ou destrói [25]. Os ataques passivos são de difícil detecção por não influírem no comportamento da rede. Nos *ataques ativos*, além de ler as informações, um atacante pode modificá-las e corrompê-las [79].

A classificação desses ataques é ilustrada a seguir na Figura 2.5, e descrita posteriormente.

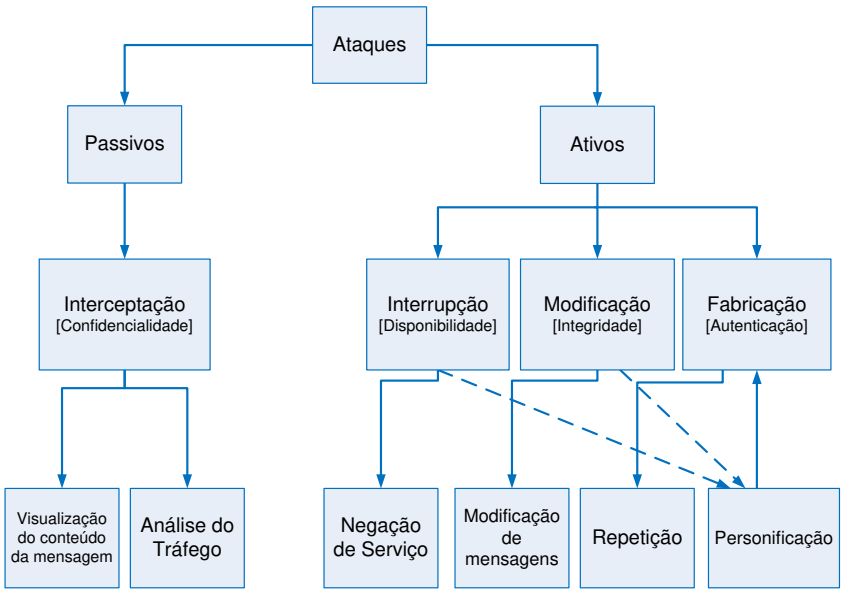

Figura 2.5: Classificação de Ataques [79].

- **Personificação** (*Masquerade/Spoofing*): enviar ou receber mensagens usando a identidade de outro, sem sua autorização;

- **Repetição** (*Replaying*): as mensagens interceptadas são armazenadas e enviadas posteriormente. Este ataque pode ser realizado mesmo quando a comunicação faz uso de autenticação e mensagens cifradas;

- **Mensagens modificadas** (*Message Tampering*): é a interceptação e alteração das mensagens antes que estas cheguem ao seu destino. O ataque do homem do meio (*man-the-middle*) é uma forma de produzir tal ataque. Nesse caso, um atacante intercepta as primeiras mensagens da troca das chaves criptográficas e as substitui por chaves conhecidas que o habilitam a decifrar as mensagens subsequentes antes de cifrá-las novamente com a chave correta e passá-las adiante;

- **Negação de serviço** (*Denial of Service*): inundação de um canal de comunicação ou outro recurso com mensagens para negar acesso a outros.

Conforme a Figura 2.5, observa-se que os ataques passivos ameaçam a confidencialidade, e são divididos de duas maneiras:

- **Visualização ou divulgação do conteúdo da mensagem:** geralmente a escuta da comunicação é feita com um *software sniffer* (programa de espionagem). Neste caso, informações importantes podem ser visualizadas por pessoas indevidas;

- **Análise de tráfego:** saber que tipo de dado e a quantidade de informação que a rede transporta, pode ser útil para ataques de negação de serviço.

Em síntese, ataques tem início com a obtenção de informações sobre os sistemas-alvo, passando por técnicas que incluem negação de serviços (*Denial of Service*), ataques ativos e ataques às aplicações e aos protocolos. Vírus, *worms* e cavalos de Troia também podem ser utilizados como um ataque ou parte dele. Tratar esses perigos implica identificar possíveis atacantes, os métodos, as técnicas e tecnologias utilizadas por eles.

2.1.4 Técnicas de Segurança

Sistemas computacionais são implantados conforme uma política de segurança. Para que eles sejam considerados seguros é necessário a adoção de técnicas de segurança. Esses mecanismos são responsáveis pela garantia das propriedades da segurança da informação.

Esse conjunto de técnicas de segurança, se bem empregado, pode contribuir para se obter sistemas distribuídos mais seguros. As principais técnicas de segurança são:

Firewall

A mais antiga definição para *firewalls* foi dada por Bill Cheswick e Steve Bellowin, em *Firewalls and Internet Security: Repelling the Wily Hacker* [17]. Segundo eles, o *firewall* é um ponto entre duas ou mais redes, no qual circunda todo o tráfego, além de registrar, por meio de *logs*, todo o tráfego da rede, facilitando sua auditoria.

Firewalls são mecanismos utilizados para proteger uma rede conectada à *Internet* [16]. Operam como uma barreira de segurança entre a rede interna e a *Internet*. Ou seja, todo tráfego de entrada e saída de dados na rede interna é analisado pelo *firewall* [34]. A Figura 2.6 o uso de *firewall* numa rede.

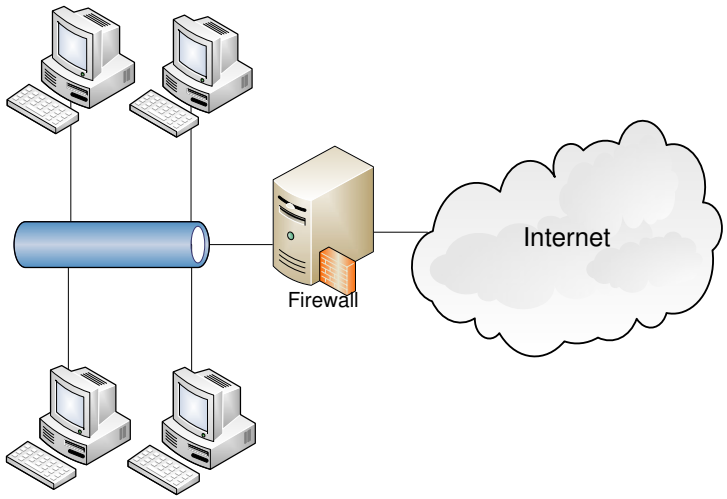

Figura 2.6: Uma rede de computadores com Firewall.

IDS, IPS e Honeypots

O *firewall* pode funcionar como primeira linha de defesa controlando e monitorando os acessos dos usuários. A autenticação para os serviços também é importante para controlar o acesso aos recursos e o sistema de detecção de intrusão (IDS) é essencial para o monitoramento do ambiente, tanto da rede quando dos servidores.

Um sistema de detecção de intrusão (*Intrusion Detection System - IDS*) tem como objetivo detectar atividades suspeitas, impróprias, incorretas ou anômalas. Além de ser crucial para a segurança interna, o IDS pode detectar ataques que são realizados por meio de portas legítimas permitidas e que, portanto, não podem ser protegidos pelo *firewall* [58].

Já um sistema de prevenção de intrusão (*Intrusion Prevention System - IPS*) tem como objetivo diminuir a quantidade de alarmes falsos e prevenir os ataques. É essencialmente um sistema de defesa baseado em rede, que combina a técnica do *firewall* com a do IDS adequadamente de forma proativa. Esse sistema que previne ataques antes de entrar na rede, examinando vários registros de dados e detecta comportamento com um sensor de reconhecimento de padrões. Quando um ataque é identificado, blocos de prevenção de intrusão são acionados e os *logs* de dados gerados [80].

Enquanto os sistemas de detecção de intrusão podem ser utilizados como fonte de aprendizado sobre novos ataques, os *Honeypots* podem ensinar muito mais. Um *Honeypot* não contém dados ou aplicações muito importantes para a organização, e seu único propósito é de passar-se por um legítimo equipamento da organização que é configurado para interagir com um invasor em potencial. Assim, os detalhes da técnica utilizada e do ataque em si podem ser capturados e estudados [58].

Forense computacional

A importância da forense computacional na investigação de crimes na *Internet* vem aumentando conforme o número e o grau de sofisticação dos ataques também aumentam. A forense computacional é a ciência multidisciplinar que tem

como objetivo o estudo de técnicas para aquisição, preservação, recuperação e análise de dados em formato eletrônico e armazenados em algum tipo de mídia [58].

Ela é importante principalmente em casos nos quais um sistema sofre algum incidente de segurança, após passar pelas defesas implementadas, por exemplo, pelo *firewall, IDS* e autenticação. Casos de fraudes financeiras, suspeitas de pedofilia, roubo de informações, confidenciais ou acessos não autorizados a sistemas críticos podem ser analisados sob a ótica da forense computacional, em busca de vestígios sobre o ataque que indiquem culpados para um possível processo judicial.

Sistemas Criptográficos

A criptografia é a ciência que tem importância fundamental para segurança da informação, ao servir de base para diversas tecnologias e protocolos, tais como a infraestrutura de chaves públicas (*Public Key Infraestruture - PKI*), o *IP Security (IPSec)* e o *Wired Equivalent*. Suas propriedades - confidencialidade, integridade, autenticação e não-repúdio - garantem o armazenamento, as comunicações e as transações seguras, essenciais na Internet hoje em dia [58].

Criptografia (do Grego *kryptós*, "escondido", e *gráphein*, "escrita") é a ciência de manter as mensagens seguras. A cifragem (*encryption*) é o processo de disfarçar a mensagem original (*plaintext* ou *cleartext*), de tal modo que sua substância é escondida em uma mensagem com texto cifeado (*ciphertext*), enquanto a decifragem (*decryption*) é o processo de transformar o texto cifrado de volta em claro original [75].

Os processos de cifragem e decifragem são realizados por algoritmos com funções matemáticas que transformam textos claros, que podem ser lidos, em textos cifrados, que são inteligíveis. Os algoritmos de criptografia dividem-se em dois grupos básicos [58]:

- **simétricos:** é responsável pelo sigilo das informações, por meio da utilização de uma mesma chave secreta para codificação e decodificação dos dados. Esses algoritmos são rápidos, porém não permitem assinatura e certificação digitais. Exemplos são o *DES, 3DES, IDEA, RC6, AES e Blowfish*.

- **assimétricos:** podem possibilitar, além do sigilo, integridade, não-repúdio e autenticidade. Requerem duas chaves (pública e privada), uma para o processo

de cifrar a mensagem e a outra para decifrar. A chave pública pode ser utilizada pelo remetente para cifrar uma mensagem, e a chave privada pode ser utilizada pelo destinatário para decifrar a mensagem. Esses algoritmos permitem assinatura e certificação digitais. Exemplos são *RSA, Diffe-Helman, ECC e El Gamal*.

Esses algoritmos se fundamentam em garantir a confidencialidade, isto é, que a informação não seja revelada ou usada indevidamente. Entretanto, certas vezes não se precisa de segredo, mas sim de integridade. Por exemplo, alguém redige um testamento para distribuir suas propriedades. O testamento não precisa ser criptografado. Após a morte do autor do documento, qualquer um poderá lê-lo. Contudo, sua integridade deve ser preservada [28].

O controle de integridade consiste em assegurar-se de que a mensagem recebida é a enviada pela outra parte e não foi manipulada. Para cumprir com este objetivo utilizam-se funções *Hash*. As funções *Hash* são conhecidas por resumirem o dado. A principal aplicação dessas funções é a comparação de dados grandes ou secretos.

As funções *Hash* são largamente utilizadas para buscar elementos em bases de dados, verificar a integridade de arquivos baixados ou armazenar e transmitir senhas de usuários. As funções *Hash* mais conhecidas são MD5 e a família *SHA (SHA-1, SHA-256 e SHA-512)* [28].

Entretanto, a autenticidade da mensagem é alcançada por meio de **Assinatura Digital**. Essa tecnologia se baseia em chave pública e vem sendo amplamente utilizada. Na assinatura digital, um resumo da mensagem (um *Hash*) é cifrado usando a chave privada do emissor. Essa assinatura é enviada junto com a mensagem.

Para verificar a assinatura, é preciso decifrar a assinatura com a chave pública do emissor e recalcular o *Hash* da mensagem. Se a comparação dos resumos (o cifrado e o calculado) for igual, a assinatura será autêntica, preservando assim a sua integridade e impedindo que posteriormente o emissor negue o envio da mensagem (não-repúdio).

Em relação ao uso de chaves públicas, um problema encontrado é justamente na origem pública deste tipo de chave. Assim, em um algoritmo RSA, por exemplo, qualquer participante pode enviar sua chave pública para outros participantes da comunidade.

Embora isso pareca conveniente, há um grande problema: qualquer um pode falsificar um aviso público. Isto é, algum usuário poderia fingir ser um usuário A, por exemplo, e enviar a sua chave pública a outros. Até que A descubra a falsificação e avise aos outros participantes, o falsificador está habilitado a ler todas as mensagens codificadas para A e usar a chave falsa para autenticação.

Uma das soluções para este problema é o **Certificado Digital** que vincula chaves públicas a principais. Para ter certeza que a chave pública no certificado pertence ao sujeito do certificado, o certificado é assinado por uma entidade confiável, chamada de Autoridade Certificadora (*Certificate Autority - CA*), que é confiável a todos os usuários da comunidade, tal como uma agência de governo ou uma instituição de finanças.

Tipicamente, em uma **Infraestrutura de Chave Pública (ICP)**, pode haver dois tipos de certificados - um certificado de nomes e um certificado de autorização. O certificado de nomes liga uma chave pública a um nome. Já o certificado de autorização associa uma chave pública a atributos e significa uma autorização ou privilégio que tem significado ao assinante e a alguns principais. Vários certificados estão em utilização. Exemplos são *o Pretty Good Privacy (PGP), SET (Security Electronic Transaction) e o Internet Protocol Security (IPSec)*. O certificado considerado mais aceito é o que segue o padrão *X.509*.

2.1.5 Padronização

Com intuito garantir que um nível adequado de segurança e das melhores práticas de segurança a serem adotadas em uma organização, vários padrões foram elaborados e fornecem estruturas (relatórios e normas) abrangentes para prática de segurança e, embora, não sejam exigidos, ajudam entidades a fazer perguntas certas, atacar as áreas devidas e estabelecer processos certos para cada circunstância.

A necessidade de estabelecer uma política de segurança é um fato realçado unanimamente em recomendações provenientes tanto no meio militar (como o *Orance Book* do Departamento de Defesa dos Estados Unidos) como do meio técnico (como o *Site Security Hadbook* 2196 do *Institute Engineering Task Force, IETF*) e, mais recentemente, do meio empresarial (norma *International Standardization Organization / International Electricaltechnical Commission (ISO/IEF)* 17799).

O artigo *"Information security management system standards: A comparative study of the big five"* [83] descreve os cinco grandes padrões de segurança da informação (*ISO 27001, BS 7799, PCIDSS, ITIL e COBIT*). O estudo realizado determinou os pontos fortes, principais componentes e orientação para fundamentar o Sistema de Gestão da Segurança da Informação (SGSI). Cada padrão desempenha seu próprio papel e posição na implementação de SGSI. Padrões como *ISO 27001 e BS 7799* focam orientações para o desenvolvimento do SGSI de uma organização, enquanto *PCIDSS* tratam a segurança da informação relacionada com as transações comerciais e inteligentes através de cartão. Por fim, *ITIL e COBIT* referem-se à segurança de informação do ponto de vista do gerenciamento de projetos e governança de TI [83].

2.1.6 Princípios de Controle de Acesso

Em sistemas distribuídos, os recursos são fornecidos para diversos usuários desconhecidos para suas aplicações. Portanto, é necessário definir o controle de acesso aos recursos, que inclui três conceitos fundamentais: autenticação, autorização e cumprimento [72, 86], conforme ilustrado na Figura 2.7, e descritos a seguir.

- **Autenticação:** especificada em termos de protocolos, abordagens de determinação de quem pode acessar o recurso protegido. É o processo de verificar se a identidade do usuário é legítima, isto é, se o usuário é realmente quem diz ser. Exemplos: senha, cartão, impressão digital, etc;
- **Autorização:** expressa em termos de um modelo de controle de acesso, a autorização especifica os recursos que precisam ser protegidos, quais tipos de acesso (operações) são possíveis aos recursos, e que acesso é permitido a eles. Por exemplo, um sistema de computador usa uma regra de controle de acesso para

decidir se é permitido ou negado o acesso a um recurso baseado na identidade obtida durante a autenticação;

- **Cumprimento:** é definida por meio da validação dos requisitos para uma entidade ser autorizada a acessar determinado recurso. Ou seja, certificar a permissão do acesso ao recurso, por exemplo, quando satisfaz a condição de um cadastro; e, a obtenção do acesso por essa entidade aos recursos quando solicitado, verificando a disponibilidade do recurso quando solicitado ou negado caso o usuário não seja autorizado.

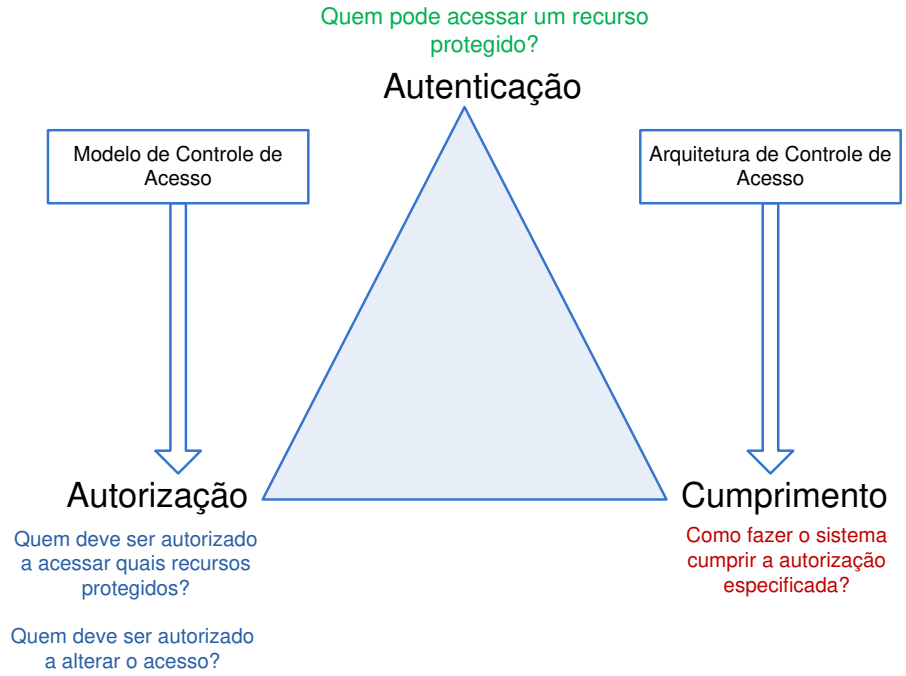

Figura 2.7: Escopo dos Conceitos de Controle de Acesso. Adaptado de [67]

Diversas terminologias para a descrição dos modelos de controle de acesso têm sido desenvolvidas durante as últimas décadas. Qualquer mecanismo de controle de acesso pode ser descrito formalmente usando descrições de usuários, sujeitos, objetos, operações e permissões, e os relacionamentos entre estas entidades. Uma breve descrição desses termos será apresentada a seguir [26]:

- **Usuários:** referem-se tipicamente às pessoas que interagem com o sistema de computador, mas também podem representar outros agentes como algum dispositivo ou uma máquina;

- **Sujeitos:** são processos ou tarefas que agem em nome do usuário e executam uma operação ou uma série de operações sobre um ou mais objetos. Dois ou mais

sujeitos podem corresponder a um mesmo usuário, mesmo que este tenha apenas um login e esteja na mesma sessão. Uma sessão é uma instância de diálogo do usuário com o sistema. Todos os sujeitos têm um identificador único;

- **Objetos:** são entidades do sistema sobre as quais as operações são executadas. No contexto de um Sistema Operacional, um objeto pode representar um arquivo. No contexto de um sistema gerenciador de banco de dados, um objeto pode ser uma tabela ou uma *view*. Outros exemplos de objetos são: *buffers*, diretórios de arquivos, páginas da web, programas, impressoras. A escolha de quais entidades que irão pertencer ao conjunto de objetos é determinada pelos requerimentos de proteção e objetivos de segurança do sistema;

- **Operações:** são processos ativos invocados pelo sujeito. Exemplos de operações no contexto de um sistema bancário podem ser: fazer depósito, saques, verificação do saldo, entre outros;

- **Permissões (ou privilégios):** são direitos dados a um individuo, ou ao sujeito agindo em nome do usuário, permitindo ao detentor do direito executar alguma ação no sistema. O termo permissão refere-se a uma relação entre o objeto, o sujeito, e a operação.

2.2 Computação em Nuvem

O termo nuvem ou *cloud* é uma metáfora em relação à forma como a Internet é usualmente mostrada nos diagramas de rede – como uma nuvem. Nesses diagramas, o ícone da nuvem representa todas as tecnologias que fazem a Internet funcionar, abstraindo a infraestrutura e a complexidade que esta engloba [91].

Já a Computação em Nuvem, refere-se de modo geral a uma combinação de tecnologias (Virtualização, Computação Utilitária, Computação em *Cluster* e Arquitetura Orientada a Serviços), em que o usuário não precisa ter todo o conhecimento necessário para manter essa infraestrutura [20].

Em [54], Mell e Grance do NIST definem a Computação em Nuvem como um modelo para permitir acesso de rede ubíquo, conveniente, e sob demanda a um repositório compartilhado de recursos computacionais (redes, servidores,

armazenamento, aplicações e serviços) que podem ser rapidamente provisionados e liberados com esforço mínimo de gerenciamento ou interação com o provedor de serviços. Para os autores, os serviços de Computação em Nuvem baseiam-se em cinco características essenciais, que são:

1. **Autoatendimento sob demanda:** um consumidor pode provisionar recursos de computação, tais como processamento e armazenamento, sem a necessidade de interação humana com cada prestador de serviço;

2. **Acesso amplo à rede:** recursos são disponibilizados sobre a rede e acessados por meio de mecanismos padronizados que permitam utilizar quaisquer tipos de plataforma, como por exemplo, telefones celulares, *tablet*, *notebooks* e estações de trabalho;

3. **Agrupamento de recursos:** os recursos são agrupados para atender múltiplos consumidores com diferentes recursos físicos e virtuais atribuídos dinamicamente à medida que a demanda do consumidor é alterada;

4. **Elasticidade:** os recursos podem ser provisionados e liberados flexivelmente (em alguns casos automaticamente), para se ajustar à necessidade crescente ou decrescente. Para o consumidor, os recursos disponíveis para provisionamento parecem ser ilimitados e podem ser consumidos em qualquer quantidade e a qualquer momento;

5. **Monitoramento de serviço:** os provedores de Computação em Nuvem controlam e aperfeiçoam, automaticamente, o uso dos recursos, aproveitando uma capacidade de medição em algum nível de abstração apropriado para o tipo de serviço. O uso de recursos pode ser monitorado, controlado, criando-se relatórios que fornecem transparência, tanto para o provedor quanto para o consumidor.

2.2.1 Modelos de Serviços

Serviços de Computação em Nuvem possuem três modelos diferentes, incluindo Infraestrutura como Serviço (IaaS), Plataforma como Serviço (PaaS) e Software como Serviço (SaaS) com base em diferentes Acordos de Nível de Serviço (SLAs) entre um provedor de serviços e um consumidor. A Figura 2.8 apresenta as camadas de serviços de Computação em Nuvem e, em seguida, a descrição de cada uma:

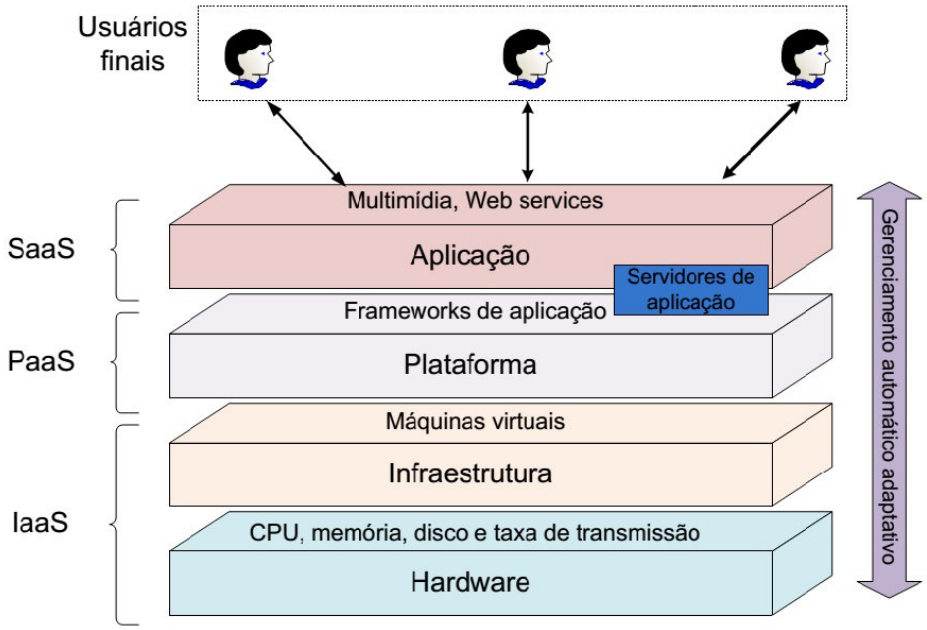

Figura 2.8: Modelos de Serviços de Nuvem. Adaptado de Zhang et al. em [98]

- **IaaS:** O consumidor contrata uma infraestrutura de hardware podendo escolher a capacidade da configuração (i. e. frequência de processamento, quantidade de memória, quantidade de armazenamento, taxa de transferência e quantidade de dados transmitidos), qual sistema operacional será executado e softwares estarão disponíveis arbitrariamente. O consumidor tem controle total da infraestrutura virtual contratada, mas não tem controle sobre a localização ou os recursos físicos de fato. É, por exemplo, o tipo de serviço oferecido pela *Amazon EC2*;

- **PaaS:** Geralmente usada para desenvolvimento colaborativo, o consumidor contrata uma plataforma com um conjunto de ferramentas específicas (ex. controle de versões, ferramentas de comunicação e disponibilização de aplicações), usando as linguagens de programação suportadas pelo provedor,

como é oferecido pela *Google Apps*, por exemplo. Nesse tipo de serviço o consumidor não tem acesso direto a configuração dos recursos físicos ou das camadas mais baixas de software;

- **SaaS:** O consumidor contrata a utilização de uma aplicação que está hospedada e em execução em uma nuvem, por exemplo, serviços oferecidos por um provedor de acesso a Internet. Esse é o tipo de serviço mais comum e, muitas vezes, o consumidor sequer tem ideia de que o serviço contratado faz parte de uma nuvem. Assim como em PaaS, nesse classe de serviço o consumidor também não tem acesso às camadas mais baixas da infraestrutura, por exemplo, o *Google Docs* e o *Windows Live Mesh*.

2.2.2 Modelos de Implantação

Baseado em Acordos de Nível de Serviço (SLA), todos os modelos de serviços de Computação em Nuvem (ou seja, IaaS, PaaS, SaaS) podem ser provisionados através de quatro diferentes modelos de implantação de serviços de Computação em Nuvem: Privada, Comunitária, Pública, e Híbrida [54, 78], dependendo das necessidades do consumidor de serviços de Computação em Nuvem.

A Figura 2.9 ilustra como serviços em nuvem estão dispostos para apoiar esses quatro serviços em nuvem a seus modelos de implementação e mostra diferentes interações entre os provedores de serviços de Computação em Nuvem e consumidores. As interações incluem *business-to-business (B2B)* e *business-to-client (B2C)*.

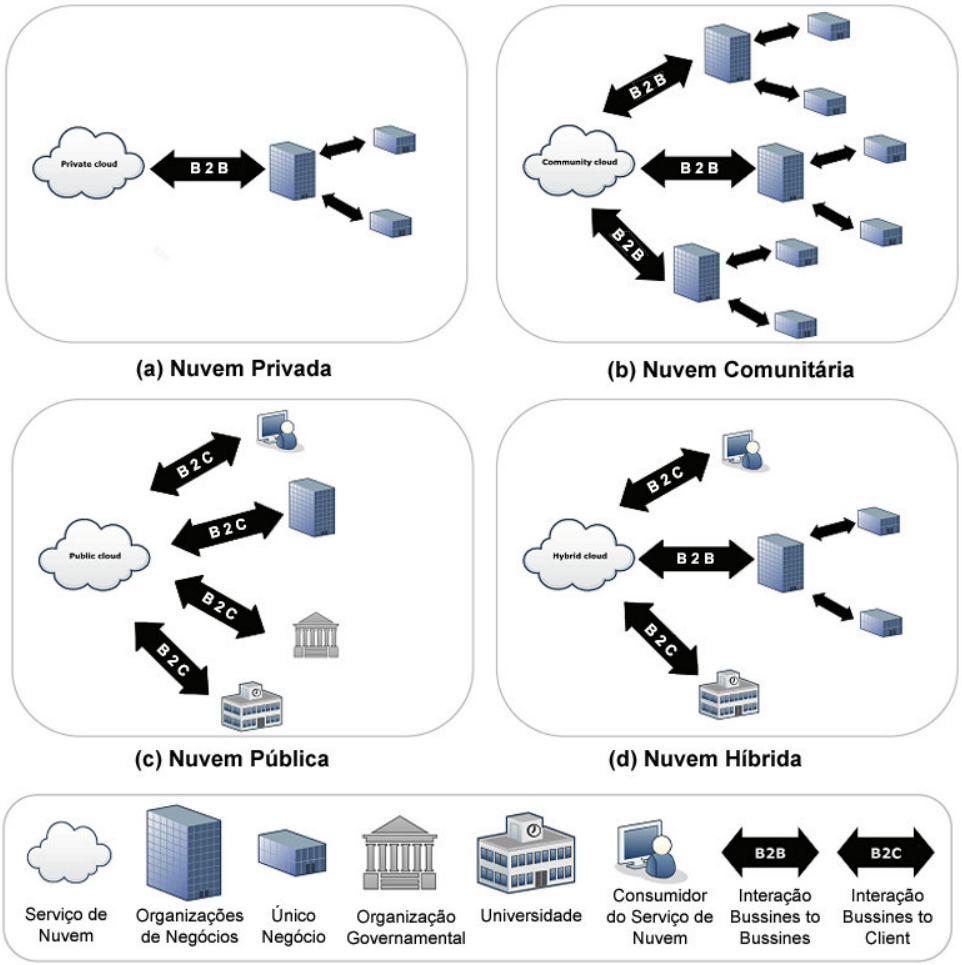

Figura 2.9: Modelos de Implantação de Nuvem. Adaptado de Noor et al. em [60]

- **Nuvem Privada:** recursos computacionais são provisionados por *uma organização particular* (uma organização de negócio como ilustra a Figura 2.9(a)), que envolve vários consumidores (várias unidades de negócios). Essencialmente,

as interações são consideradas como B2B em que os recursos computacionais podem ser pertencidos, gerenciados e operados pela mesma organização, um terceiro, ou ambos.

- **Nuvem Comunitária:** recursos computacionais são provisionados por *uma comunidade de organizações* como ilustra a Figura 2.9(b), para atingir um determinado objetivo (por exemplo, alto desempenho, requisitos de segurança, ou redução de custos). Basicamente, as interações são consideradas como B2B, onde os recursos de computação podem pertencer, ser gerenciados e operados pela mesma comunidade, um terceiro, ou ambos.

- **Nuvem Pública:** recursos computacionais são provisionados pelo *público* (por exemplo, um consumidor individual de serviços de nuvem, universidade, governo, organizações de empresas , ou uma combinação desses tipos de consumidores de serviços de nuvem, como mostrado na Figura 2.9(c)). Essencialmente, as interações são consideradas como B2C em que os recursos computacionais podem pertencer, ser gerenciados e operados por uma universidade, governo, ou organização de empresas, ou uma combinação de dois desses.

- **Nuvem Híbrida:** recursos computacionais são provisionados por *dois ou mais modelos de implantação de nuvem* (por exemplo, nuvens privadas e públicas podem ser implantadas em conjunto, utilizando um modelo de implantação híbrido, como mostrado na Figura 2.9(d)). Basicamente, as interações incluem B2B e B2C, em que os recursos computacionais são interligados por diferentes nuvens utilizando técnicas de portabilidade (por exemplo, portabilidade de dados e aplicações, e em casos de nuvens cujos recursos estão saturados, necessitando de balanceamento de carga entre nuvens).

2.2.3 Segurança em Nuvem

Segundo os autores [50, 68, 82, 90] existem diversos aspectos críticos de segurança em nuvem. Os principais são:

- **Segurança de Identidades**: objetiva manter a integridade e confidencialidade dos dados e das aplicações, ao mesmo tempo que tornam o acesso disponível aos clientes autorizados. Os principais padrões e soluções utilizadas para a implantação de gerenciamento de identidades incluem o *SAML (Security Assertions Markup Language), OpenID, Shibboleth, Higgins, SimpleSAMLphp, OpenAM, XACML (Extensible Access Control Markup Language) e OAuth*;
- **Segurança da Informação**: nos tradicionais data centers, controles de acesso físico, de acesso ao *hardware* e *software* e de identidade combinam-se para proteger os dados. Na nuvem, a barreira de proteção da infraestrutura é difusa. Assim, a segurança da informação exigirá [11]:
  - **Isolamento dos dados:** na multi-alocação de recursos os dados do ambiente devem ser armazenados de forma segura, a fim de protegê-los quando vários clientes utilizam os recursos compartilhados. Virtualização, criptografia e controle de acesso são mecanismos que permitem vários graus de isolamento de dados entre empresas e clientes diferentes;
  - **Segurança dos dados:** os provedores de serviços devem fornecer mecanismos de segurança para proteger os dados de seus clientes. Isso envolve o uso de técnicas de criptografia e controle de acesso aos dados;
  - **Segurança de rede:** todo o fluxo de dados da rede deve estar seguro para evitar a perda e manipulação de informações. Para isso, podem-se utilizar técnicas de criptografia ou outras técnicas que garantam a segurança e o monitoramento do tráfego de rede;
  - **Integridade dos dados:** qualquer tipo de transação deve seguir as propriedades ACID (Atomicidade, Consistência, Isolamento e Durabilidade) para garantir a integridade dos dados;
  - **Vulnerabilidade na virtualização:** uma máquina virtual oferece um ambiente completo similar a uma máquina física. Dessa forma, algumas vulnerabilidades encontradas em uma máquina física são também

encontradas nos softwares de virtualização. Essas vulnerabilidades podem ser utilizadas por invasores para adquirirem determinados privilégios e violar outras restrições de segurança.

- **Segurança de Infraestrutura**: quando uma empresa contrata um serviço de IaaS e presta serviços a outros clientes, o provedor de IaaS é indiferente quanto às operações e ao gerenciamento de pedidos das empresas contratantes do seu serviço. Assim, é importante que o contratante assuma total responsabilidade por assegurar a sua infraestrutura, implantando, por exemplo, mecanismos de controle de acesso, de criptografia dos dados e/ou ferramentas de monitoramento da rede. Além disso, a segurança da infraestrutura física de um provedor deve ser garantida.

2.3 Gerenciamento de Identidades

Nas relações comerciais, provedores de serviços (loja virtual, sistema acadêmico ou portal) necessitam de algum nível de confiança entre as partes pra realizar transações. Quando essa confiança não pode ser estabelecida diretamente, a interação ocorre por meio de um intermediário confiável para ambas as partes (também chamado de terceira parte), conforme ilustrado na Figura 2.10. Essa terceira parte é responsável por fornecer informações sobre o parceiro, dando origem ao conceito de identificação.

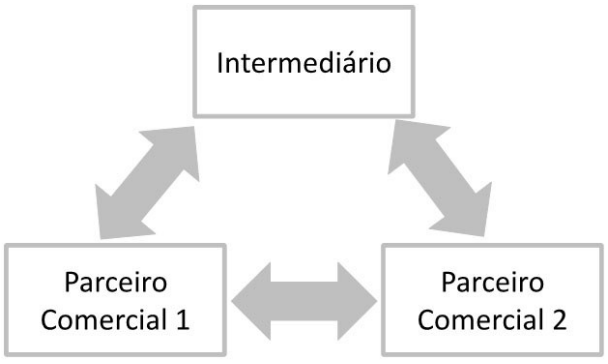

Figura 2.10: Relação comercial envolvendo um intermediário [89]

As identidades das partes envolvidas na transação são controladas e monitoradas por *Sistemas de Gerenciamento de Identidades (SGI)*. Esses são programas ou

frameworks que administram uma coleção de identidades, realizam sua autenticação, gerenciam seu uso e as informações vinculadas à identidade [22].

Sendo assim, o gerenciamento de identidades consiste de um sistema integrado de políticas, processos de negócios e tecnologias que permite às organizações o tratamento e a manipulação de identidades (atributos de identidade) de seus usuários [12, 43]. O gerenciamento de identidades também envolve aspectos relacionados com a definição, certificação e gerenciamento do ciclo de vida das identidades digitais, infraestruturas para troca e validação dessas informações, juntamente com os aspectos legais [43].

2.3.1 Componentes

O processo de identificação pode ser compreendido a partir dos seguintes componentes da identidade: *sistema, entidade, identidade, identificador e credencial*, conforme Torres et al. em [89].

- **Sistema:** é um conjunto de software e hardware que armazena informações sobre entidades, credenciais, identidades, e processamento das operações de autenticação e autorização para proteger dados e serviços. Ex: *Shibboleth, OpenAM, SimpleSAMLphp, OpenID* etc;
- **Entidade:** pode ser um pessoa, um serviço de rede, um dispositivo computacional em rede ou um dispositivo de telefonia móvel. As entidades existem no mundo real. Elas usam credenciais e possuem um ciclo de vida independente;
- **Identidade:** é um conceito virtual utilizado por uma entidade para fornecer informações sobre si para o sistema. Uma identidade sempre está associada a uma entidade, geralmente constituída de um identificador único. O identificador é utilizado para provar a propriedade da identidade (por meio de credenciais) e fornecer informações (perfil) a um sistema. O sistema irá utilizar essas informações para tomar decisões sobre a entidade associada;

- **Atributos:** um conjunto de dados que descreve as características fundamentais de uma identidade. Como exemplo tem-se: nome completo, domicílio, data de nascimento e papéis [41];

- **Identificador:** é o índice único de uma identidade. Normalmente, um identificador é usado pelo sistema para referenciar uma identidade. Por exemplo, uma URL (Uniform Resource Locator) é única ao longo do tempo. Como exemplo de identificadores temos CPF, RG, número de matrícula e número de passaporte;

- **Credencial:** serve para provar uma identidade em um sistema. Podem existir vários tipos de credenciais, mas todas são utilizadas para provar a um sistema (com um nível aceitável de segurança). Exemplos de credenciais incluem certificados digitais *X.509* assinados por uma autoridade certificadora (*CA - Certificate Authority*), senha, asserções *SAML (Security Assertions Markup Language)*, dentre outros.

A ilustração do relacionamento entre os componentes da Identidade, é vista na Figura 2.11 em notação *UML (Unified Modeling Language)*.

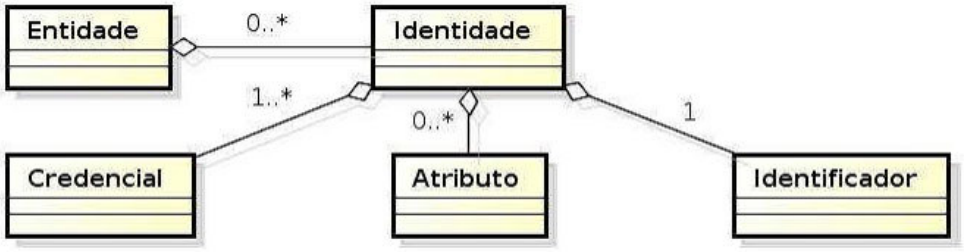

Figura 2.11: Relacionamento entre os componentes da Identidade [24]

2.3.2 Operações

As principais operações no gerenciamento de identidades são [89]:

- **Identificação:** é a função de uma entidade fornecer uma identidade ao sistema por meio de um identificador;

- **Autenticação:** é a função do sistema verificar a legitimidade de uma identidade por meio da verificação de credenciais;

- **Autorização:** é a função do sistema conceder privilégios a uma entidade após a autenticação da sua identidade;

- **Auditoria:** é a função de registrar as ações realizadas por uma entidade em um sistema, gerando uma prova tanto para as partes envolvidas quanto para terceiros.

2.3.3 Ciclo de Vida

A Figura 2.12 ilustra a ciclo de vida de uma identidade e a relação entre as suas fases. O ciclo de vida de uma identidade pode ser dividido [8]:

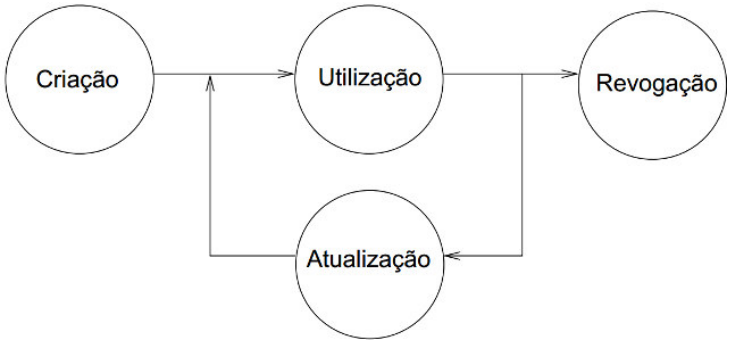

Figura 2.12: Ciclo de Vida da Identidade [24]

- **Criação:** a criação de uma identidade decorre de tecnologia utilizada. Sistemas automatizam sua criação pra armazenar informações de usuários em um repositório (por exemplo, banco de dados). Essa etapa passa pela verificação de atributos (por exemplo, RG e CPF), precedida da emissão da credencial (por exemplo, certificados digitais e senhas) e, por fim, a formação da identidade constituída de seus atributos, credenciais e identificadores atribuídos por terceiros ou pelo usuário;

- **Utilização:** depois de criada, a identidade pode ser utilizada por vários sistemas e serviços. O uso da identidade implica na utilização de forma segura e privativa;

- **Atualização:** refere-se à alteração de valores de atributos já existentes (por exemplo, um novo endereço) ou a inserção de novos atributos para refletir novas políticas ou regras de negócio;

- **Revogação:** identidades e credenciais devem ser canceladas quando ficarem obsoletas (por exemplo, expirou a data de validade) ou inválidas.

2.3.4 Atores

Em um Sistema de Gerenciamento de Identidades, podem-se identificar três atores [9]:

- **Usuário:** aquele que acessa algum serviço através de seus atributos (por exemplo, nome e endereço);
- **Provedor de Identidades ou Identity Provider (IdP):** responsável por emitir a identidade de um usuário. Após o usuário passar por um processo de autenticação, este recebe uma credencial, dita identidade, que é reconhecida como válida pelos provedores de serviço;
- **Provedor de Serviços ou Service Provider (SP):** oferece recursos a usuários autorizados, após verificar a autenticidade de sua identidade e após comprovar que a mesma carrega todos os atributos necessários para o acesso.

A interação entre esses três atores do sistema de gerenciamento de identidade é vista na Figura 2.13, e descrita em seguida.

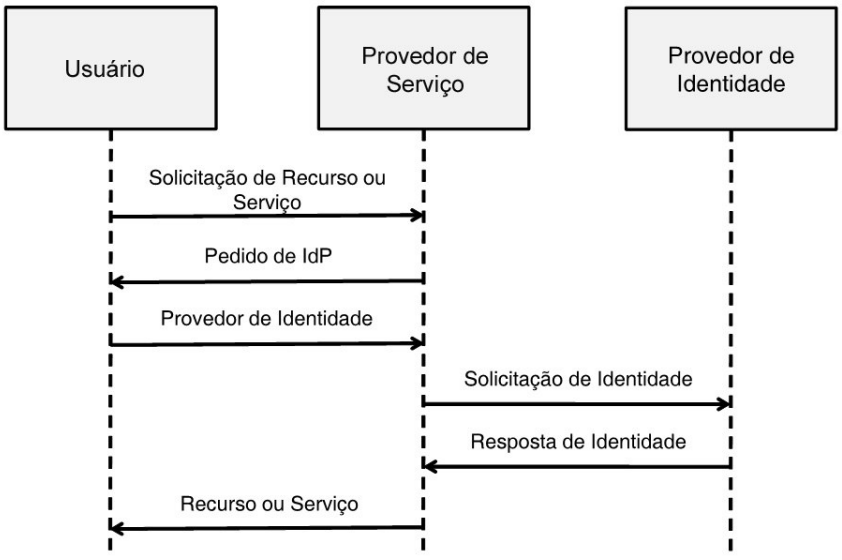

Figura 2.13: Atores em sistemas de gerenciamento de identidades. Adaptado de [89]

Conforme Figura 2.13, o usuário solicita um serviço a um provedor de serviços que, por sua vez, confia em um provedor de identidades, o qual é responsável pelo fornecimento de informações de autenticação do usuário.

2.3.5 Modelos

Os modelos de gerenciamento de identidades apresentam diferentes formas de interação, dependendo de como os atores citados anteriormente estão dispostos. Os Sistemas de Gerenciamento de Identidades seguem os seguintes modelos (conforme ilustra a Figura 2.14 e descrito em seguida) [42, 43]:

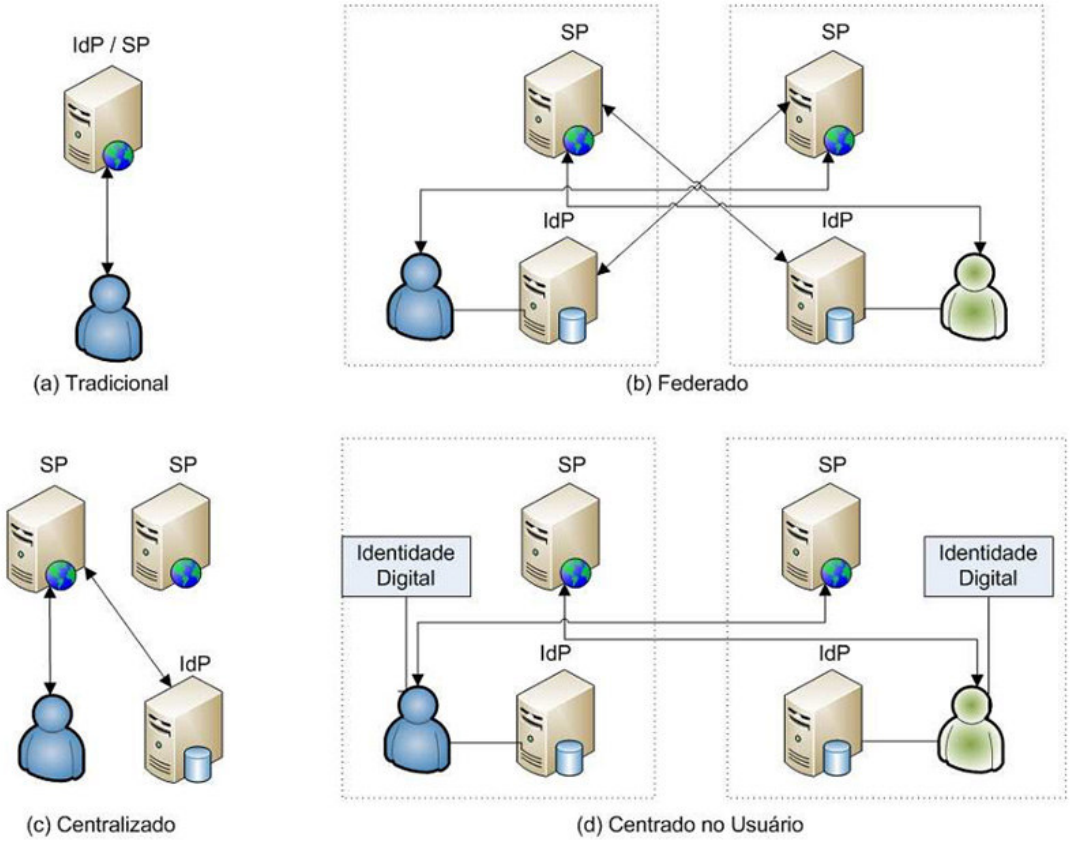

Figura 2.14: Modelos de Gerenciamento de Identidades [24]

No modelo **Tradicional**, a identificação do usuário é tratada de forma isolada por cada provedor de serviços, que ao mesmo tempo atua como provedor de identidades, conforme Figura 2.14(a). Uma desvantagem deste modelo é a gestão de múltiplas senhas por usuários para interagir com provedores (por exemplo, *e-mails*, sítio de notícias, etc);

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 (*Single Sign On – SSO*) [9]. Exemplos de sistemas centralizados incluem *Kerberos*, *Public Key Infraestructure - PKI*, *Central Authentication Service - CAS* e *Microsoft Passport Network*.

O modelo **Federado** é uma evolução do modelo centralizado. Nesse modelo, os SPs e o IdP podem estar em domínios administrativos diferentes. No modelo federado é possível também existir vários IdPs, com uma relação MxN entre IdPs e SPs. Um domínio administrativo pode representar uma empresa, uma universidade, etc e é composto por usuários, diversos provedores de serviços e um único provedor de identidades. Exemplos de sistemas federados incluem o projeto *Kantara* e *Shibboleth*, descritos na Seção 2.4.1.

Os provedores de serviço confiam no modelo de gerenciamento de identidade das instituições e disponibilizam seus serviços para os usuários vinculados a essas instituições, criando assim o princípio de *identidade federada*. Portanto, uma federação é tipo de rede de confiança que permite reduzir contratos bilaterais entre usuários e provedores de serviços. A implementação de identidade federada por ser exemplificada quando instituições mesmo utilizando métodos de autenticação diferentes, ainda sim mantém a interoperabilidade entre elas.

O modelo **Centrado no usuário**, uma evolução do modelo federado, visa resolver um dos principais desafios fornecidos pelos modelos anteriores, que refere-se à identidade do usuário, que fica armazenada nos provedores de identidades. Nesse caso, os IdPs podem ter controle total sobre estas informações. Portanto, o modelo centrado no usuário oferece mecanismos para que um usuário possa escolher que tipo de informações deseja liberar para um determinado provedor de identidades. Exemplos de soluções que utilizam o modelo centrado no usuário são: *OpenID*, *Microsoft CardSpace* e o Projeto *Higgins*.

2.3.6 Identidades Federadas em Computação em Nuvem

Nesta pesquisa, o termo "Identidades Federadas" refere-se ao modelo Federado. Esta abordagem de gerenciamento de identidades pode ser considerada a mais adequada para Computação em Nuvem [51]. A *Cloud Security Alliance - CSA*

recomenda o desenvolvimento de arquiteturas que utilizem a Identidade como um Serviço (IDaaS) [68].

Tormo em [88] analisa os padrões atuais de gerenciamento de identidade (por exemplo, SAML, OpenID, OAuth etc) sob a perspectiva da Computação em Nuvem, considerando as suas características e necessidades. E Lonea et al. em [51] descreve quatro requisitos de gerenciamento de identidades para Computação em Nuvem: provisionamento/desprovisionamento de identidades, autenticação, identidade federada e controle de acesso.

Esses autores fornecem recomendações quanto à concepção e implantação de qualquer serviço baseado em identidade em Computação em Nuvem. As funções de gerenciamento de identidades aplicadas à Computação em Nuvem são [88]:

- ***Single Sign On* Federado e Compartilhamento de Atributos:** permite que os usuários finais acessem múltiplos serviços com um único conjunto de credenciais, enquanto a segunda refere-se ainda a esses serviços que também podem exigir a recuperação de informações dos usuários para prover outros serviços ou para realizar alguma função de controle de acesso;

- **Agregação de Atributos e Operações:** os usuários finais geralmente têm seus atributos distribuídos por múltiplas fontes de informação (universidade, prefeitura, banco, hospital, etc), mantidos por diferentes provedores em diferentes domínios. Eles ainda podem efetuar operações sobre os atributos de forma a apresentar apenas as informações necessárias para acessar um serviço;

- **Privacidade de Identidade em Ambientes Compartilhados:** provedores de serviços de nuvem geralmente exigem atributos de usuários finais antes de fornecerem o próprio serviço na nuvem (por exemplo, serviços de compras *on-line* exigem o endereço postal para enviar os itens comprados). Esses atributos são considerados confidenciais, ou seja, apenas as partes confiáveis devem ter acesso a eles.

2.4 Trabalhos Relacionados

Nos últimos anos, diversos trabalhos da literatura científica e industrial têm se apresentado com o objetivo de integrar identidades federadas para compartilhar credenciais de acesso na infraestrutura de Computação em Nuvem. Esta seção está divida em três partes. A primeira descreve as principais iniciativas de normalização de gerenciamento de identidades, a segunda apresenta as abordagens de integração de identidades, enquanto a terceira mostra pesquisas relacionadas à integração de identidades federadas em ambiente de Computação em Nuvem, que serviram de base para elaboração desta dissertação.

2.4.1 Iniciativas de normalização

I) Órgãos de Padronização

O ***OASIS (Organization for the Advancement of Structured Information Standards)*** é um órgão de normalização bem conhecido por estruturar padrões de informação. Especificações e protocolos como o *SAML, UDDI, WS-*, AMQP, XACML, XDI* são desenvolvidos pela *OASIS*.

A ***Identity Commons*** é uma associação de grupos em comunidade para o desenvolvimento de camada de identidade centrada no usuário na Internet, que permite ao usuário controlar a sua privacidade de maneira conveniente. Cada grupo é responsável por um objetivo específico.

O ***ITU-T (Telecommunication Standardization Sector)*** é um organismo de normalização para o setor de Telecomunicações. *ITU-T Identity Management Focus Group* facilita o desenvolvimento e a padronização de um *framework* genérico de gerenciamento de identidades.

II) Projetos

O projeto ***PRIME (Privacy and Identity Management for Europe)*** [66] aborda a falta de infraestrutura de gerência de identidade para a Internet, identificando suas principais carências, tais como segurança e privacidade, e visando definir um equilíbrio dos requisitos emergentes para os SGIs. O objetivo do projeto consiste em desenvolver um protótipo de trabalho para melhorar a privacidade dos SGIs,

permitindo gerenciar as múltiplas identidades que um consumidor pode obter através de suas interações pela Internet. Exemplos de interações são as operações comerciais realizadas pela Internet [4].

O ***DAIDALOS (Designing Advanced network Interfaces for the Delivery and Administration of Location independent, Optimised personal Services)*** [19] foi um dos primeiros projetos voltados para o gerenciamento de identidades fortemente centrado nas infraestruturas de redes e serviços. Um dos resultados desse projeto é o conceito de Identidade Virtual (VID) e sua aplicação em tais infraestruturas, em que o VID denota a entidade em uma função específica ou contexto de uso, e não a entidade em sua totalidade [74].

O ***SWIFT (Secure Widespread Identities for Federated Telecommunications)*** é um projeto Europeu, financiado pelo programa FP7, do inglês *7th Framework Programme*. O projeto aproveita a tecnologia de gerenciamento de identidades para integrar o serviço e a infraestrutura de transporte a fim de beneficiar usuários e prestadores de serviços. O objetivo é estender as funções de gerenciamento de identidades e a federação para redes sem negligenciar a usabilidade e privacidade [89].

III) ALIANÇAS

A iniciativa ***Kantara*** é uma evolução da *Liberty Alliance* estabilizada em 2008 para criar um ponto central e robusto dentro da comunidade de identidades. O objetivo desta aliança é a reunião de trabalhos em questões-chave, incluindo a interoperabilidade e conformidade de testes, garantia de identidade, política de privacidade e de desenvolvimento de software. Os membros da iniciativa, que contribuem para seu financiamento, incluem o Projeto *DataPortability*, o Projeto *Concordia*, *Liberty Alliance*, a *Internet Society (ISOC)*, Fundação Cartão de Informação (ICF), *OpenLiberty.org* e *XDI.org* [45].

A ***FIDIS (Future of Identity in Information Society)*** [27] é uma Rede de Excelência (do inglês, *Network of Excellence* ou NoE) suportada pela União Européia. Essa aliança integra esforços de pesquisa das diferentes nações européias buscando solucionar problemas desafiadores, tais como suporte à identidade e à identificação, interoperabilidade de identidades, conceitos de identificação, roubo de identificadores, privacidade, segurança e perfis de usuários, além de buscar soluções para problemas

forenses. Uma das principais atividades de pesquisa da aliança está focada na melhor definição de identidade.

IV) Especificações Consolidadas

Padrões de Autenticação

O ***SAML (Security Assertions Markup Language)*** resulta dos trabalhos do OASIS [62] *Security Services Technical Committee*, mas a versão atual do SAML conta com uma grande contribuição da iniciativa *Kantara*. O SAML é um padrão XML para a troca de dados de autenticação e autorização entre serviços. Esse padrão XML resolve dois problemas: *Single Sign On* ("Autenticação única") e identidades federadas, e geralmente é destinado a parceiros comerciais que buscam um padrão para troca segura de informações [56].

O ***OpenID*** [64] é um *framework* aberto, descentralizado e gratuito voltado para a identidade digital do usuário. OpenID permite aos usuários de Internet acessar diferentes sites, baseados em uma única identidade digital, o que elimina a necessidade de diferentes nomes de usuário e senha para cada sítio web [64]. Seu desenvolvimento iniciou em 2005 e obteve expressiva atenção, tanto da comunidade de desenvolvimento Web quanto de grandes corporações. Hoje, *Google, IBM, Microsoft, VeriSign e Yahoo* são membros da Fundação *OpenID*.

O ***Shibboleth*** é uma iniciativa originada no ambiente acadêmico, cujo objetivo é facilitar o compartilhamento de recursos de pesquisa entre instituições acadêmicas. Ele é um padrão que tem como base pacotes de *software* de código aberto com acesso único para web entre ou dentro dos limites organizacionais. Esse padrão permite que sites tomem decisões de autorização para acesso individual a fim de proteger recursos online, de forma a preservar a privacidade. O *software Shibboleth* implementa abordagens de identidade federada para fornecer um acesso único federado e um *framework* de troca de atributos.

Padrões de Autorização

XACML (Extensible Access Control Markup Language) é um padrão que define uma linguagem declarativa, utilizada para a descrição de políticas de controle de acesso, e uma linguagem para formular consultas (e obter respostas) utilizada para verificar se uma determinada ação é ou não permitida. Como resposta à consulta, um

dos seguintes valores pode ser retornado: *Permit, Deny, Indeterminate* (um erro ocorreu, ou algum valor não foi especificado, impedindo o prosseguimento da consulta), ou *Not Applicable* (quando a resposta não puder ser retornada pelo serviço solicitado).

OAuth é um padrão aberto que fornece um mecanismo para que um usuário/aplicação possa compartilhar recursos na Web com terceiros sem ter que compartilhar a sua senha [63]. Dessa forma, um usuário proprietário de um recurso (um álbum de fotos, por exemplo) pode oferecer acesso temporário a esse recurso sem compartilhar a sua identidade (nome de usuário e senha). Atualmente, há um projeto de trabalho em andamento do *OAuth 2.0*, que está sendo suportado por várias empresas, como *Facebook, Google e Microsoft*.

2.4.2 Abordagens

Esta seção tem como objetivo descrever as abordagens existentes de *"integração de identidades"*, termo que assume a mesma conotação de *"agregação de atributos"* no contexto desta dissertação.Além disso, essa seção discute e compara algumas pesquisas de integração de identidades.

Os primeiros trabalhos sobre integração de identidades assumem que o usuário tenha um identificador único comum entre todas as autoridades de atributos. Esse identificador é um nome X.500 contido em um certificado X.509, emitido por uma autoridade certificadora. Nessa abordagem o usuário precisa se autenticar uma única vez com seu certificado de chave pública, para que o processo de agregação dos atributos possa ser executado. Porém, atualmente, poucos usuários possuem certificados X.509. É mais comum que esses tenham diferentes nomes de usuário e diferentes atributos em seus vários IdPs [14].

Na literatura científica, podem-se encontrar seis abordagens para integração de identidades em sistemas de gerenciamento de identidade. São elas [36,46]:

1. **Banco de dados da aplicação:** a forma mais simples de integração de identidades permite que o SP colete as informações sobre o usuário nos IdPs a partir de atributos informados pelo usuário, ficando por conta do SP gerenciar os atributos adicionais do usuário;

2. ***Proxying* de identidade:** as comunidades formam organizações virtuais (do inglês *Virtual Organizations - VO*). Essas organizações mantêm as informações sobre seus membros. Um IdP proxy transforma ou estende uma identidade obtida por um IdP em uma identidade que contém as informações necessárias ao SP. O IdP *proxy*, neste caso, deve ser confiável tanto para o SP que depende deste, mas também pelos IdPs que disponibilizam os atributos ao IdP *proxy*.

3. ***Proxy* de retransmissão de identidade:** a técnica de retransmissão de identidade pode ser considerada uma especificação de um *proxy*. Embora as trocas de atributos no modo de retransmissão de identidade seja semelhante ao modelo de IdP *proxy*, no modelo de retransmissão tem-se a garantia das asserções de atributos. Nesse caso, o IdP proxy não irá assinar as asserções de atributos, mas sim retransmiti-las no formato original assinadas pelo IdP original. O SP irá receber as afirmações de atributos criptografadas;

4. **Agregação de atributos mediada pelo cliente:** aproveita-se do uso de clientes inteligentes (clientes ativos) que podem criar solicitações, mantendo o estado e agregando informações sobre o usuário. Exemplos desses clientes são o *ECP SAML* (do inglês E*nhanced Client or Proxy*) e os clientes ativos do *CardSpace*;

5. **Federação de identidade:** depende da capacidade do usuário em associar as identidades que controla. Um agente do usuário autenticado com sucesso em duas entidades diferentes pode controlar ambas as entidades e criar um identificador unidirecional persistente, permitindo que um IdP aponte para a identidade relacionada no segundo IdP. Isso pode ser repetido pelo provedor para criar uma ligação bidirecional. É necessária a utilização de asserções para associar as informações da identidade do usuário. Para permitir que terceiros utilizem esse link de associação e agregar os atributos do usuário, o mapeamento precisa ser incorporado em uma afirmação. *SAML* e *ID-WSF* (do inglês *Identity Web Services Framework*) possibilitam a incorporação de valores adicionais nas afirmações.

6. **SP mediando a agregação de atributos:** dependendo da capacidade do SP, é possível que este seja o mediador da agregação de atributos e isso é feito pela obtenção de informações de vários IdPs. Apesar da autenticação do usuário ser única para cada IdP, o SP realizará excessivos redirecionamentos para

completar a agregação de atributos. Esta solução é uma das mais simples de ser desenvolvida, mas possibilita o rastreamento das informações dos usuários, visto que o SP irá mediar todo o processo de agregação de atributos.

2.4.3 Pesquisas em Identidades em Nuvem

O trabalho de Leandro et al.em [49] apresenta um modelo de gerenciamento de identidades federadas para acesso a aplicações de Computação em Nuvem. Um cenário é estruturado de forma que os serviços (implantados em máquinas virtuais) são fornecidos pela *Amazon EC2*, e provedores de autenticação em um ambiente de terceiros, separados e baseados em *Shibboleth* e *Central Authentication Service (CAS)*.

Em [81], Stihler et al. fornece uma arquitetura para integração de identidades federadas focada em usuários IaaS, com o objetivo de suportar múltiplos domínios em ambientes de múltiplos provedores. Nesse cenário, ocorre a tradução transparente do alto nível de identidades (por exemplo, em nível de SaaS), permitindo que usuários efetuem a autenticação de seus IdPs, interajam com o SaaS e usufruam da operação *Single Sign-On - SSO*. A construção da arquitetura incluiu componentes tecnológicos como o IdP *OpenID*, *SAML* e o provedor de nuvem *Eucalyptus*.

Chadwick et al. em [13] integra o provedor de nuvem *Openstack* com IdPs usando o protocolo *SAML*. O IdP utilizado foi o *SimpleSAMLphp*, que implementa o protocolo *SAML* e possui um componente chamado "*Proxy IdP*" que faz interface com outros provedores de identidade nos padrões *SAML, OAuth, OpenID*, dentre outros.

O mesmo autor ainda, em [15], implementa um modelo para integrar dois protocolos diferentes, o SAML e o *Keystone* proprietário da plataforma *OpenStack*. O *Keystone* é o serviço de identidade que autentica usuários, fornecendo um *token* de autorização aos diversos serviços *OpenStack*. E, em [87], Silva et al. apresenta um estudo de caso para a integração do serviço *Swift* do *Openstack* com a autenticação federada por meio do IdP *Shibboleth*.

A Tabela 2.1 apresenta uma análise comparativa entre os trabalhos descritos anteriormente.

Tabela 2.1: Comparação de Trabalhos

(Autor)	Ano	IdP Utilizado	Usuário de Nuvem	Plataforma
Leandro et al [49]	2012	Shibboleth, CAS	SaaS	Amazon EC2
Stihler et al [81]	2012	OpenID	IaaS, PaaS, SaaS	Eucalyptus
Chadwick et al [13]	2013	SimpleSAMLphp	IaaS	Openstack
Silva et al [87]	2013	Shibboleth	IaaS	Openstack
Chadwick et al [15]	2014	IdPSAML	IaaS	Openstack

2.5 Considerações Finais

Este capítulo apresentou os conceitos fundamentais para a elaboração desta pesquisa. Inicialmente, a seção *Segurança da Informação* descreveu os principais termos relacionados à segurança da informação, técnicas de segurança existentes e os princípios de controle de acesso.

Quanto a isso, observou-se que as maiores ameaças estão nas vulnerabilidades resultantes de falhas na implementação dos produtos (sistemas operacionais, protocolos, aplicativos), nas configurações equivocadas dos sistemas e na engenharia social (técnica que explora as fraquezas humanas e sociais, um exemplo clássico consiste em se fazer passar por um funcionário que tem problemas urgentes de acesso ao sistema).

Na seção *Computação em Nuvem*, foi mostrada uma visão geral deste paradigma, destacando características, modelos e seus aspectos de segurança. Depois, na seção *Gerenciamento de Identidades*, foram apresentados os conceitos e abordagens gerais para implantação de sistemas de gerenciamento de identidades. Nesse contexto, percebeu-se que a criptografia, em suas diversas facetas, tem muito a contribuir para a segurança de sistemas distribuídos, auxiliando nos sistemas de autenticação.

Esse estudo conduziu ainda à percepção de que o uso de um gerenciamento de identidades depende do tipo de modelo de nuvem: para SaaS, apenas o controle de acesso às aplicações é necessário; para PaaS e IaaS, é necessário o controle de acesso integrado tanto das aplicações quanto do sistema e demais recursos (*software* e *hardware*) acessíveis na plataforma. Assim, mais pesquisas devem ser realizadas intencionando melhorar a privacidade e a interoperabilidade dos mecanismos de controle de acesso aos recursos da nuvem.

Por fim, na seção *Trabalhos Relacionados*, foram descritos os principais órgãos de padronização e demais iniciativas que incluem projetos, alianças e padrões consolidados relativos ao gerenciamento de identidades. Foram também discutidas pesquisas recentes sobre a integração de sistemas de gerenciamento de identidades federadas em ambiente de Computação em Nuvem, fundamentais para elaboração desta pesquisa.

3 Modelo Proposto

Este capítulo apresenta uma visão geral do *Modelo de Integração de Identidades Federadas para Aplicações de Computação em Nuvem Baseado em Proxy* e uma descrição do seu funcionamento (fluxos de comunicação de dados).

3.1 Visão geral do modelo

O modelo proposto neste trabalho é descrito como um mecanismo de autenticação federada que utiliza asserções *SAML* responsáveis pelo compartilhamento das informações de identidade do usuário entre os diferentes provedores de identidade (IdPs) que dão acesso às aplicações (serviços oferecidos pelo SP) contidas no ambiente de Computação em Nuvem, conforme ilustrado na Figura 3.1.

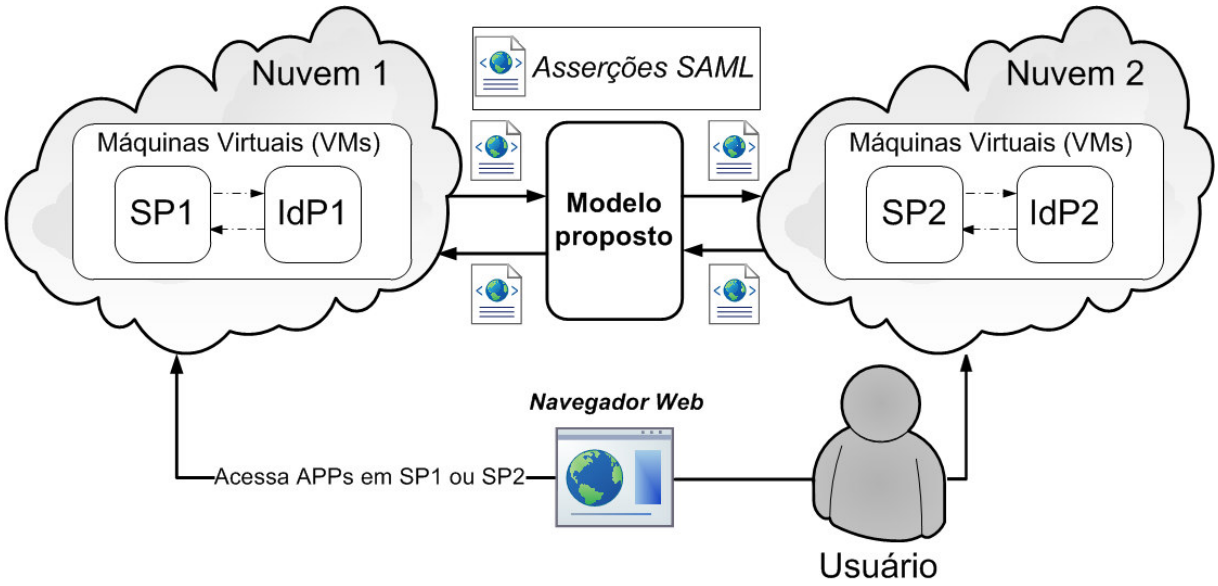

Figura 3.1: Visão geral do Modelo Proposto.

Este modelo utiliza os princípios de identidades federadas e as características de implantação de um ambiente de Computação em Nuvem, cuja atividade principal é a realização do *Single Sign-On (SSO)* ou "Autenticação única". Conforme o modelo ilustrado na Figura 3.1, o acesso aos serviços (aplicações ou APPs)

oferecidos pelos provedores de serviços (*SP1* e *SP2*) por meio de uma única senha cadastrada em qualquer IdP (*IdP1* ou *IdP2*).

Uma parte dessa ideia é vista no trabalho apresentado por Morgam et al. [56], que descreve o sistema de identidades federadas e atributos do *Shibboleth* para sistemas baseados em *Web*. Na pesquisa desse autor, o *Shibboleth* é desenvolvido em conformidade com as especificações SAML, consistindo de provedores de serviço (consumidores de assertivas) e provedores de identidade (fornecedores de assertivas).

Enquanto Huang et al. [35] propõem, em nível de *SaaS*, a utilização de uma terceira parte confiável com papel de intermediador (*broker*). Nesse contexto, um determinado cliente pode contratar serviços de diferentes provedores, mas mantendo as identidades federadas dos usuários. A pesquisa desse autor implementada o *IdPSAML*, porém não explicita uma plataforma de nuvem em seu experimento.

Com base nisso, considerou-se a possibilidade de utilizar identidades federadas para integrar aplicações de diferentes nuvens por meio do protocolo *SAML* para compartilhar credenciais de acesso entre provedores distintos de identidade, incluindo, neste contexto de pesquisa, os *frameworks Shibboleth* e *SimpleSAMLphp*. Portanto, o problema central a ser superado por esta pesquisa é que cada sistema de gerenciamento de identidades utiliza uma solução própria para administrar as identidades e controlar o acesso às aplicações da nuvem.

A *Cloud Security Alliance* [68] recomenda que os provedores de serviços devem ser flexíveis e que aceitem os formatos utilizados pelos provedores de identidades. Argumenta também que as organizações não devem integrar diretamente as aplicações oferecidas em nuvem para realizar *Single Sign-On (SSO)* e o acesso a recursos.

Federações de identidade permitem que a autenticação dos usuários seja realizada por entidades que possuem relacionamentos próximos aos mesmos, por exemplo: universidade-alunos/professores/pesquisadores, empresas-colaboradores, banco-cliente, grandes provedores de serviço na Internet-usuários [76].

O uso de federações também permite que o usuário se autentique uma única vez e obtenha acesso a vários serviços e sistemas diferentes através do mecanismo de *Single Sign-On (SSO)*. Por fim, ao unificar a autenticação, os usuários passam a ter menos senhas para lembrar, podendo escolher senhas mais fortes e seguras [69].

Além disso, garantir a privacidade é um grande desafio no compartilhamento de informações de identificação pessoal (atributos de identidade). Em sistemas de identidades federadas, o compartilhamento seguro de tais informações é essencial para se usufruir dos benefícios da federação.

Visando impedir a rastreabilidade dos atributos de identidade do usuário (garantir a privacidade), possível nas soluções baseadas em *proxy*, o mecanismo agregador de atributos proposto neste trabalho segue a abordagem mediada pelo cliente, e que segue o modelo de retransmissão de asserções *SAML*, com o diferencial de ser alinhado ao ambiente de nuvem, fornecendo o suporte à autenticação SSO.

O procedimento de integração de identidades é executado pelas ferramentas e o protótipo implementado inseridos da camada PaaS das plataformas de nuvem utilizadas no experimento. O mecanismo de integração de identidades gerencia o pseudônimo, conforme provido pela especificação SAML, para aumentar a privacidade do usuário e dificultar o rastreamento de suas informações nos provedores de serviços.

3.2 Requisitos

Segundo [38,46], os doze requisitos considerados importantes pelo usuários para implementação de um modelo para integração de identidades, podem ser observados na Tabela 3.1.

Tabela 3.1: Requisitos do Usuário para Integração de Identidades. Adaptado de [38,46]

Categoria	***Requisitos***
Gerais	**1)** A integração de identidades pode ser utilizada por uma variedade de formas, como: por seres humanos, com uso do navegador Web; por aplicações, por meio das *APIs (Application Programming Interface)*; por usuários de rede, por intermédio dos clientes de rede.
Privacidade	**2)** A proteção da privacidade dos atributos do usuário é importante e pode ser feita por meio do uso de técnicas de controle que são independentes dos meios legais. **3)** Os provedores de serviço podem ser capazes de rastrear os usuários entre suas sessões, se necessário. **4)** Em circunstâncias excepcionais, os provedores de serviço podem identificar a verdadeira identidade dos usuários, no entanto, somente solicitando a identificação junto aos IdPs do usuário.
Consentimento (*User*)	**5)** IdPs e SPs podem se comunicar para unir os atributos do usuário e este processo deve ser de conhecimento e aprovação do usuário. **6)** Somente com a aprovação dos usuários, os SPs podem consultar seus múltiplos IdPs, a fim de agregar os atributos adicionais dos usuários.
Exigências do Protocolo	**7)** Os protocolos (preferencialmente os protocolos *HTTP* e *HTTPS*) devem ser capazes de funcionar na presença de *firewalls*, utilizando-se de portas de comunicação já existentes. **8)** O protocolo *SAML* é a escolha mais usual em sistemas de gerenciamento de identidades federadas. **9)** O *proxy* de informação deve ser suportado por meio de múltiplos saltos (*redirects*).
Confiança	**10)** A capacidade opcional para assinar todas as asserções deve ser suportada em todas as trocas de mensagens. **11)** O SP deve ser capaz de exigir que todas as asserções sejam assinadas pelas autoridades de atributos.
Usabilidade	**12)** O sistema deve ser fácil de utilizar e exigir o mínimo de interação do usuário.

3.3 Cenário

A Figura 3.2 apresenta um cenário que envolve todas as entidades descritas anteriormente. Neste cenário, o contratante IaaS obtém do contrato os recursos providos por diferentes plataforma de nuvem (IaaS 1 e 2), neste caso, Máquinas Virtuais - VMs. Depois, o IaaS contratante implanta sua aplicação SaaS sobre o recursos contratados, e se beneficia da administração da infraestrutura e da aplicação. Além disso, o estabelecimento das relações de confiança com outros provedores de identidade (IdPs) se faz necessário para autenticação dos usuários das aplicações SaaS.

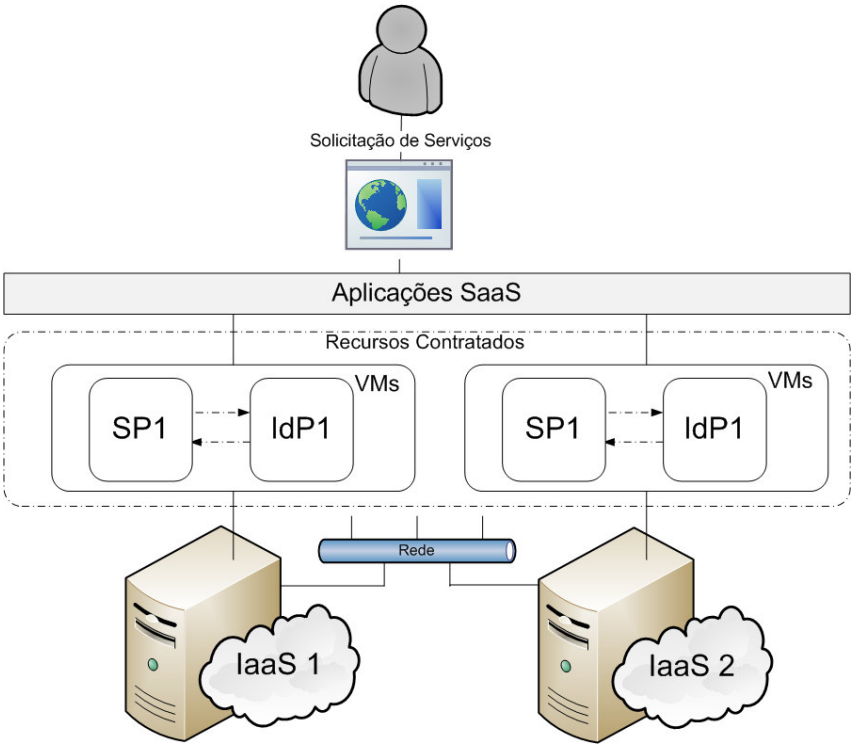

Figura 3.2: Cenário Exemplo do Modelo Proposto.

Na prática, quando o usuário efetua a autenticação em um dos IdPs (IdP1, por exemplo), imediatamente é autorizado ao consumo da aplicação (no SP1) no IaaS 1 cuja VM em que está localizado o serviço de autenticação (IdP1). Quando o usuário tentar acessar outra aplicação que, desta vez, pertença a outra IaaS (no SP2), o modelo proposto baseado em *proxy* dispara a informação de que o usuário está ativo (autenticado) e transmite suas credenciais de autenticação via asserções *SAML* para o IdP2, que por sua vez, libera o acesso ao serviço da outra plataforma de nuvem (IaaS 2), propiciando ao usuário o benefício do *Single Sign On (SSO)*.

3.4 Descrição do protocolo de autenticação *SAML*

O protocolo *SAML*, criado em 2001, baseia-se na troca de mensagens de autenticação e autorização no padrão XML entre usuários e provedores de serviço e de identidades. É um protocolo com a sua última versão (2.0) lançada em 2005. Essa versão suporta vários mecanismos de transporte por meio de diferentes *bindings* [62], por exemplo: *SOAP* (única opção na versão 1.x), *HTTP Redirect (GET)* e *HTTP POST*.

O uso do *HTTP* sem o *Simple Object Access Protocol - SOAP* tornou o protocolo mais leve, apesar das mensagens ainda serem *XML*, contendo certificados e assinaturas digitais. Além dos *bindings*, a nova versão permite diferentes fluxos de autenticação, ou *profiles* [62]. No perfil *Web Browser SSO*, toda troca de dados entre IdP e SP passa pelo cliente por meio de mensagens assinadas, não havendo tráfego entre SP e IdP. Já no perfil *Artifact Resolution*, SP e IdP interoperam diretamente para trocar dados de autenticação.

A Figura 3.3 ilustra o fluxo de trabalho do *SAML*, que representa a interação entre o usuário, o provedor de serviço e o provedor de identidade.

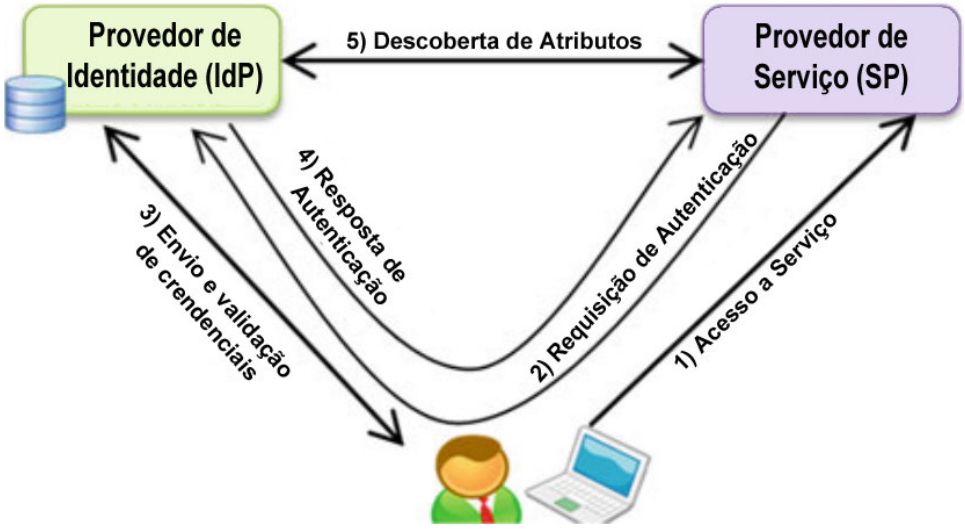

Figura 3.3: Fluxo de trabalho do SAML. Adaptado de [88]

Inicialmente, um usuário final, por meio do navegador Web, solicita acesso a um serviço de um provedor de serviços. Depois, o provedor de serviços redireciona o usuário final para seu provedor de identidade, juntamente com um pedido de autenticação *SAML*. O provedor de identidade solicita ao usuário final suas credenciais (geralmente, nome de usuário e senha).

Uma vez que o provedor de identidade valida a autenticação, ele redireciona o usuário final para o provedor de serviços, juntamente com uma asserção *SAML*, indicando que o usuário final foi autenticado. Nesse ponto, o provedor de serviços pode requerer alguns atributos do usuário final enviando mensagens de consulta a atributos *SAML* diretamente ao provedor de identidade.

3.5 Análise Comparativa

O trabalho de Leandro et al. [49] apresenta um modelo de gerenciamento de identidades federadas para acesso a aplicações SaaS de Computação em Nuvem. Nessa abordagem, os usuários possuem seus provedores de identidades fora da nuvem. Cada provedor de identidades define um domínio de políticas com os seus clientes e provedores de serviços (provedores de computação em nuvem). A associação de confiança que garante a transposição de uma asserção SAML de autenticação de um domínio para o outro é estabelecida pelo *Shibboleth*. Já no modelo proposto nesta pesquisa, os usuários possuem seus provedores de identidades dentro da nuvem por um cenário com serviços virtualizados em diferentes IdPs.

A ideia de Stihler et al. [81] é a mesma adotada pelo modelo proposto - suportar múltiplos domínios em ambientes de múltiplos provedores. Entretanto, diferencia-se por fornecer uma arquitetura para integração de identidades federadas focada para usuários IaaS, enquanto neste trabalho é focada a integração de identidades federadas para usuários SaaS. Nesses cenários, ocorre a tradução transparente do alto nível de identidades (por exemplo, em nível de SaaS), permitindo que usuários efetuem autenticação de seus IdPs, interajam com o SaaS e usufruam da operação *Single Sign-On - SSO*.

Chadwick et al. [13] integra o provedor de nuvem *Openstack* com IdPs usando o protocolo *SAML*. O IdP utilizado foi o *SimpleSAMLphp*, que implementa o protocolo *SAML* e possui um componente chamado "*Proxy IdP*" que faz *interface* com outros provedores de identidade nos padrões *SAML, OAuth, OpenID*, dentre outros.

Em [15], o mesmo autor implementa um modelo para integrar dois protocolos diferentes, o SAML e o *Keystone* proprietário da plataforma *OpenStack*. O *Keystone* é o serviço de identidade que autentica usuários, fornecendo um *token*

de autorização aos diversos serviços *OpenStack*. Em Silva et al. [87] é apresentado um estudo de caso para integração do serviço *Swift* do *Openstack* com autenticação federada através de IdP *Shibboleth*.

Como se observa, em geral, os trabalhos citados têm focado na integração de identidades em usuários IaaS de nuvem, estabelecendo a premissa de federação entre as camadas de serviços do provedor de nuvem. Este trabalho direcionou-se ao uso de integração de identidades federadas, focado em usuários SaaS pertencentes a múltiplos domínios de nuvem. Além disso, apresentou, experimentalmente, a interação entre as tecnologias de IdPs (*Shibboleth* e *SimpleSAMLphp*) aliadas ao fornecimento de *Single Sign-On (SSO)* em nuvens, um aspecto não tratado por outros autores.

A Tabela 3.2 apresenta uma análise comparativa entre o modelo aqui proposto e os trabalhos descritos anteriormente.

Tabela 3.2: Análise Comparativa com Trabalhos Relacionados

(Autor)	**Ano**	**IdP Utilizado**	**Usuário de Nuvem**	**Plataforma**
Leandro et al. [49]	2012	Shibboleth, CAS	SaaS	Amazon EC2
Stihler et al. [81]	2012	OpenID	IaaS, SaaS	Eucalyptus
Chadwick et al. [13]	2013	SimpleSAMLphp	IaaS	Openstack
Silva et al. [87]	2013	Shibboleth	IaaS	Openstack
Chadwick et al. [15]	2014	IdPSAML	IaaS	Openstack
Modelo proposto	2014	Shibboleth, SimpleSAMLphp	SaaS	Eucalyptus

3.6 Considerações Finais

Neste capítulo foi feita uma descrição sucinta do modelo proposto neste trabalho para integração de identidades federadas, utilizado para prover acesso a aplicações de nuvens. Esse modelo depende do funcionamento das tecnologias e ferramentas por ele utilizadas para poder validar as ideias aqui propostas.

A funcionalidade do modelo é apresentada sem fazer um aprofundamento nas técnicas e tecnologias utilizadas (vistas no Capítulo 4). Dessa forma, espera-se contribuir para o entendimento geral do modelo proposto. Após a descrição das funções e requisitos, é apresentada uma análise comparativa com os trabalhos citados no Capítulo 2.

O próximo capítulo apresenta uma sequência de testes e os resultados obtidos com o funcionamento do modelo, por meio de um protótipo implementado. Esses testes atentam principalmente para a capacidade do modelo de integrar os atributos federados dos IdPs utilizados como pré-requisito de autenticação para o acesso às aplicações SaaS em múltiplos domínios de nuvens.

4 Implementação e Resultados

Este capítulo apresenta a implementação, testes e resultados experimentais. Estas fases concluem o desenvolvimento da solução proposta que consiste na integração de identidades federadas entre IdPs para acesso às aplicações SaaS de múltiplos domínios de nuvens.

4.1 Observações gerais

Foram realizados alguns testes com o protótipo visando a obtenção de resultados experimentais. O modelo proposto foi implementado sobre um ambiente de máquina virtual, o que facilitou sua implantação em plataforma de nuvem. Os testes realizados tiveram por objetivo principal verificar a capacidade de detecção dos IdPs por seus respectivos *IdPsProxy* em cada sistema de gerenciamento de identidade federada utilizado no experimento.

Analisando os resultados coletados, foi possível notar a homologação dos atributos de identidade, decorrentes da primeira autenticação, cujas asserções de seu IdP de origem sendo compartilhadas para outro IdP pertencente a outro domínio de nuvem, confirmando assim o sucesso da autenticação única entre os diferentes sistemas de gerenciamento de identidade.

4.2 Ferramentas e Tecnologias Utilizadas

A fase de implementação do modelo leva em conta a definição da linguagem de programação, arquitetura de software (*frameworks*, *containers* de aplicação, bibliotecas entre outros componentes) e ferramentas de apoio ao desenvolvimento, necessárias à transformação da modelagem (definidos no capítulo 3) em um protótipo. Essa etapa envolve ainda a configuração de ambiente no qual o sistema será executado.

Esse conjunto de tecnologias deve ser adequado ao desenvolvimento da solução proposta, levando em consideração seus requisitos. O protótipo em questão é um mecanismo de integração de identidades que visa especificamente à autenticação única federada em nuvens, cujo princípio é:

1. Autenticação e captura de atributos do usuário por meio de asserções *SAML*;
2. E transmissão dos atributos às aplicações SaaS (SPs) de outra nuvem, fora do domínio de origem no qual ocorreu a primeira autenticação, também por meio de asserções *SAML*.

Logo, a escolha das ferramentas de desenvolvimento considera a viabilidade delas para o contexto de nuvem e em relação à possibilidade de interoperarem por meio do mesmo protocolo.

A seguir é apresentada uma breve descrição das ferramentas e tecnologias que foram utilizadas na implantação da infraestrutura de gerenciamento de identidade federadas para validação do modelo proposto para integração de atributos de usuários em múltiplos domínios de nuvem.

EUCALYPTUS

O *Eucalyptus* é uma infraestrutura de software de código aberto que utiliza diversos recursos computacionais disponíveis aos pesquisadores, como *clusters* e *workstation farms*. A fim de promover a exploração da comunidade científica de sistemas de computação em nuvem, o *Eucalyptus* foi desenvolvido totalmente modularizado, permitindo aos pesquisadores realizarem experimentos de segurança, escalabilidade, confiabilidade e implementação de interfaces [61].

O desenvolvimento do *Eucalyptus* foi guiado para atingir duas metas principais:

- **Liberdade:** Ser capaz de ser implantado e executado em ambientes de hardware e software que não estão sob o controle de seus criadores;
- **Extensibilidade:** deve ser modularizado, para permitir a fácil manutenção de um componente ou até mesmo sua substituição.

A extensibilidade é garantida ao construir uma arquitetura utilizando *Web Services*, na qual as interfaces são bem definidas utilizando documentos *WSDL* e a

autenticação é feita via mecanismos *WS-Security*. Ao mesmo tempo, o *Eucalyptus* suporta uma interface com a nuvem popular da *Amazon AWS*. O *Eucalyptus* é compatível com várias distribuições de *Linux*, incluindo *Ubuntu*, *Red Hat Enterprise Linux (RHEL)*, *CentOS*, *SUSE Linux Enterprise Server (SLES)*, *openSUSE*, *Debian* e *Fedora* e com uma variedade de tecnologias de virtualização, como *VMware, Xen e KVM hypervisors*.

KVM (KERNEL-BASED VIRTUAL MACHINE)

A virtualização possibilita a execução de diversas máquinas virtuais em uma mesma máquina física, o que pode trazer benefícios como flexibilidade, segurança e facilidade de configuração e gestão. O gerenciamento das máquinas virtuais é feito pelos hipervisores que controlam o acesso das máquinas virtuais ao hardware.

O *KVM (Kernel-based Virtual Machine)* é um virtualizador do tipo virtualização completa com extensões de virtualização (Intel VT ou AMD-V). Assim, o KVM consiste em um módulo de *kernel (kvm.ko)*, que fornece a infraestrutura de virtualização e um módulo de processador específico, *intel.ko kvm* ou *amd.ko kvm* [1]. Com o KVM é possível executar múltiplas máquinas virtuais, em paralelo, sem a necessidade de modificar as máquinas convidadas. Cada máquina virtual possui um conjunto de hardware virtual privado: uma placa de rede, disco, placa gráfica, etc.

SHIBBOLETH

É utilizado por federações acadêmicas de diversos países, incluindo a Comunidade Acadêmica Federada (CAFe) do Brasil. É um conjunto de softwares *Open Source* mantido pelo consórcio Internet2. Esse *framework* implementa o padrão *SAML* da *OASIS*. O conjunto é formado por dois componentes, o Shibboleth IdP e o Shibboleth SP. A versão mais atual do *Shibboleth* é, a versão 2.5.2 para o *Shibboleth SP* e a versão 2.5.1 para o *Shibboleth IdP*, porém, neste trabalho utilizou a versão 2.1.5 do *Shibboleth IdP* e a versão 2.4.3 para o *Shibboleth SP*.

Entretanto, para que a Infraestrutura de Autenticação e Autorização (IAA) por meio do *Shibboleth* funcione corretamente, são necessários os seguintes programas:

- **Apache Web Server:** é um dos mais populares e robustos servidores Web do mercado, que se faz importante para o funcionamento do ambiente federado. É utilizado por IdP e SP para estabelecer a troca de mensagens HTTP;

- **OpenSSL:** é uma implementação *Open Source* do *Secure Software Layer (SSL)* e do *Transport Layer Security (TLS)*, que têm como objetivo utilizar bibliotecas de criptografia de dados. O *Shibboleth* gera certificados e assinaturas das asserções *SAML* únicos para o IdP e SP, visando garantir a privacidade dos dados do usuário que trafegam pela Internet;

- **Apache Tomcat:** outro popular servidor Web, que tem como objetivo prover a execução de aplicações Java (*servlets* e jsp) na Internet. É útil especialmente pelo IdP, e às vezes no desenvolvimento de algum serviço ou aplicação Web que se pretenda disponibilizar por meio do SP;

- **OpenJDK:** é um projeto da *Java Virtual Machine (JVM)* para criação de um *Java Development Kit* baseado totalmente em software livre e de código aberto. Nesse contexto, dar suporte ao funcionamento do IdP *Shibboleth*, e em conjunto com o Apache Tomcat possibilita que a aplicação seja executada no navegador;

- **OpenLDAP:** é um padrão aberto que define um método para acessar e atualizar informações em um diretório. Serviços de diretório desempenham um papel importante no desenvolvimento de aplicações intranet e Internet, permitindo o compartilhamento de informações sobre usuários, sistemas, redes, serviços e aplicações. O uso do *OpenLDAP* é necessário para armazenar e gerenciar a base de usuários.

SIMPLESAMLPHP

O *SimpleSAMLphp* [2] é uma implementação de referência do protocolo *SAML*, podendo atuar como IdP ou SP. Uma vantagem adicional de utilizá-lo é o fato de que, por ser escrito em linguagem PHP (suportado por *Linux, UNIX, Mac OSX, Windows*), mostra que o protocolo *SAML* (e o modelo proposto desenvolvido) é independente de linguagem de programação (SP escrito em PHP, trocando mensagens em XML com um IdP escrito em Java). E sua configuração *IdP SimpleSAMLphp* possibilita realizar a autenticação (*LDAP, Mysql, Radius*).

4.3 Ambiente de Desenvolvimento

Conforme ilustrado na Figura 4.1, o ambiente usado para a realização do experimento é composto por dois computadores de processador Core i5 de 2.67 GHz, 4GB de memória e disco de armazenamento de 500GB, onde foram instaladas as plataformas de nuvem *Eucalyptus* na versão 3.3.1 sobre o sistema operacional Linux CentOS 6. Os computadores estão instalados no Laboratório de Sistemas em Arquiteturas Computacionais (LABSAC) da UFMA, conectado à RNP por meio de um link de 10 Gbps.

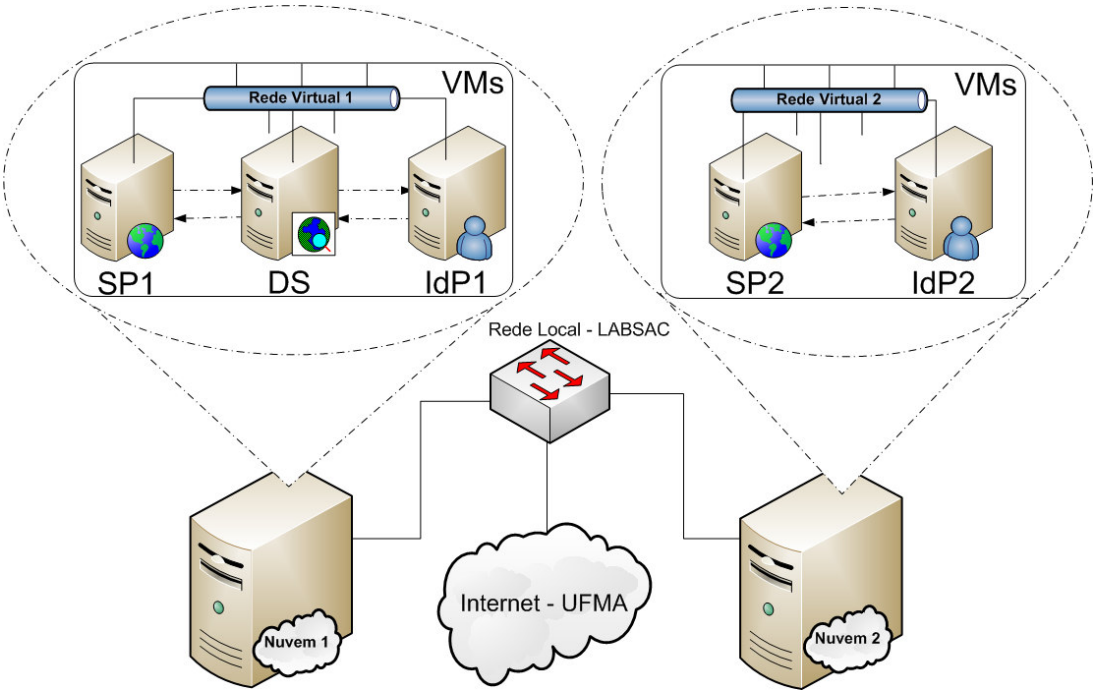

Figura 4.1: Ambiente de Desenvolvimento dos Testes

Na Nuvem1, há 3 máquinas virtuais instaladas utilizando-se o *software KVM (Kernel-based Virtual Machine)*, cada uma contém o SP, o IdP e o DS *(Discovery Service)* do *framework* de gerenciamento de identidade - o *Shibboleth*. E na Nuvem 2, têm-se 2 máquinas virtuais também *KVM*, nas quais executam o IdP e o SP do outro framework de gerenciamento de identidade - o *SimpleSAMLphp*.

Cada componente IdP1, SP1 e DS do *Shibboleth* é composto por uma VM (Máquina Virtual) com 1GB de RAM e 2GB de armazenamento, exceto o DS que tem 5GB de disco. Todos utilizam o processador Intel i5 de 2.67 GHz e sistema operacional Linux Ubuntu Server 12.04 LTS. Já para o IdP2 e SP2 do *SimpleSAMLphp*, têm-se as

mesmas configurações de hardware, exceto à capacidade de armazenamento que é de 8GB, o sistema operacional utilizado é o Linux Ubuntu 13.04.

A Tabela 4.1 mostra as configurações de rede utilizadas para o estabelecimento da comunicação de dados entre as nuvens para validação do modelo proposto. Para trafegar os dados oriundos das VMs pela rede local, foi configurada uma faixa de IPs válidos locais com acesso à Internet, de 192.168.27.241 a 192.168.27.250 para as VMs da Nuvem 1 e de 192.168.27.231 a 192.168.27.240 para as VMs da Nuvem 2. Por fim, o DNS 192.168.20.10 utilizado nas VMs é o mesmo das máquinas físicas.

Tabela 4.1: Configurações de Rede do Ambiente de Testes

Dispositivo	**Endereço IP**	**Máscara de Sub-rede**	**Gateway**
Host-Nuvem1	192.168.27.83	255.255.255.0	192.168.27.1
Host-Nuvem2	192.168.27.84	255.255.255.0	192.168.27.1
VM-SP1	10.10.10.2	255.255.255.0	10.10.10.1
VM-DS	10.10.10.3	255.255.255.0	10.10.10.1
VM-IdP1	10.10.10.4	255.255.255.0	10.10.10.1
VM-SP2	10.10.20.2	255.255.255.0	10.10.20.1
VM-IdP2	10.10.20.3	255.255.255.0	10.10.20.1

4.4 Implementação do Protótipo

Esta seção apresenta o protótipo que é uma aplicação do modelo proposto no Capítulo 3. Este protótipo tem por objetivo validar os requisitos e funcionalidades propostas pelo modelo desenvolvido, favorecendo a posterior execução de testes e a obtenção de resultados experimentais.

4.4.1 Arquitetura

Os usuários utilizam *Sistemas de Gerenciamento de Identidade* para acessar serviços que exigem a certificação de seus atributos por uma terceira parte. Isso é comum em acessos que envolvem a Internet, pois os usuários não confiam na segurança da transmissão de dados. Por isso, o gerenciamento de identidade tem como objetivo aumentar a confiança em processos que envolvem a Internet, e no caso desta

pesquisa, proteger os dados de acesso dos usuário em relação às aplicações da nuvem por ele requisitados.

No contexto deste trabalho são consideradas as seguintes entidades:

- **Usuário App:** qualquer pessoa que deseje acessar as aplicações Web da nuvem;
- **Usuário IaaS:** as entidades autorizadas a acessar os recursos IaaS (por exemplo, processamento, disco e rede);
- **Contratante PaaS:** alguém que os contratos de recursos de provedores de IaaS, para implantar as aplicações SaaS para serem disponibilizadas aos usuários;
- **Aplicação:** é o recurso SaaS ou um conjunto de serviços Web hospedados nos provedores de serviços. Podem incluir sites, portais, e-commerce, e-learning etc. Representa o serviço que o *Usuário App* poderá acessar;
- **Provedor de identidade (IdP):** qualquer entidade que controle as credenciais do usuário e forneça serviços de autenticação. Nesta proposta, é importante considerar que o usuário efetua as solicitações em nível de SaaS.
- **Provedor de Serviço (SP):** uma entidade que oferece um ou mais serviços possíveis de serem acessados pelos usuários e utiliza os serviços de uma IdP para autenticar um usuário.

O provedor de identidade (IdP) tem o papel de fornecer um grau de identificação, que inclui, por exemplo, autoridades certificadoras. Esta função visa estabelecer um Círculo de Confiança (*Circle of Trust - CoT*) exigido por algum tipo de negócio *on-line* ofertado pelo provedor de serviço (SP). Essa cadeia de segurança, o *CoT*, em geral, é integrada por servidores de autenticação ou criptografia SSL, por exemplo.

A relação de confiança entre IdP e SP é importante por contribuir para o aumento da segurança de organizações. Além disso, é importante para ambas as partes terem certeza sobre a autenticidade do parceiro de comunicação. Especialmente quando se pode evitar fraudes utilizando este meio, visando assim à garantia da privacidade das informações dos usuários.

Com base nessas premissas, propõe-se um modelo de integração de identidades federadas capaz de mapear e fornecer credenciais de IdPs com objetivo

de propiciar o acesso, por um única autenticação, às aplicações em um ambiente de múltiplas nuvens. Alguns componentes são operados na camada PaaS para esconder os detalhes da implementação, e incluem funções como a captura e replicação das credenciais do usuário.

Essas funções do modelo são essenciais para o usuário acessar qualquer aplicação de quaisquer provedores de serviço, de forma transparente e sem causar alterações na estrutura do protocolo *SAML*, e efetuar o *Single Sign-On (SSO)* em nuvens.

A Figura 4.2 apresenta os principais componentes da arquitetura desenvolvida. Os componentes *Usuário App, Interceptador e Serviço* efetuam suas operações em nível de SaaS, embora a camada PaaS forneça a infraestrutura base para o funcionamento dos demais componentes que incluem o *Provedor de Autenticação* e o *Provedor de Credenciais*.

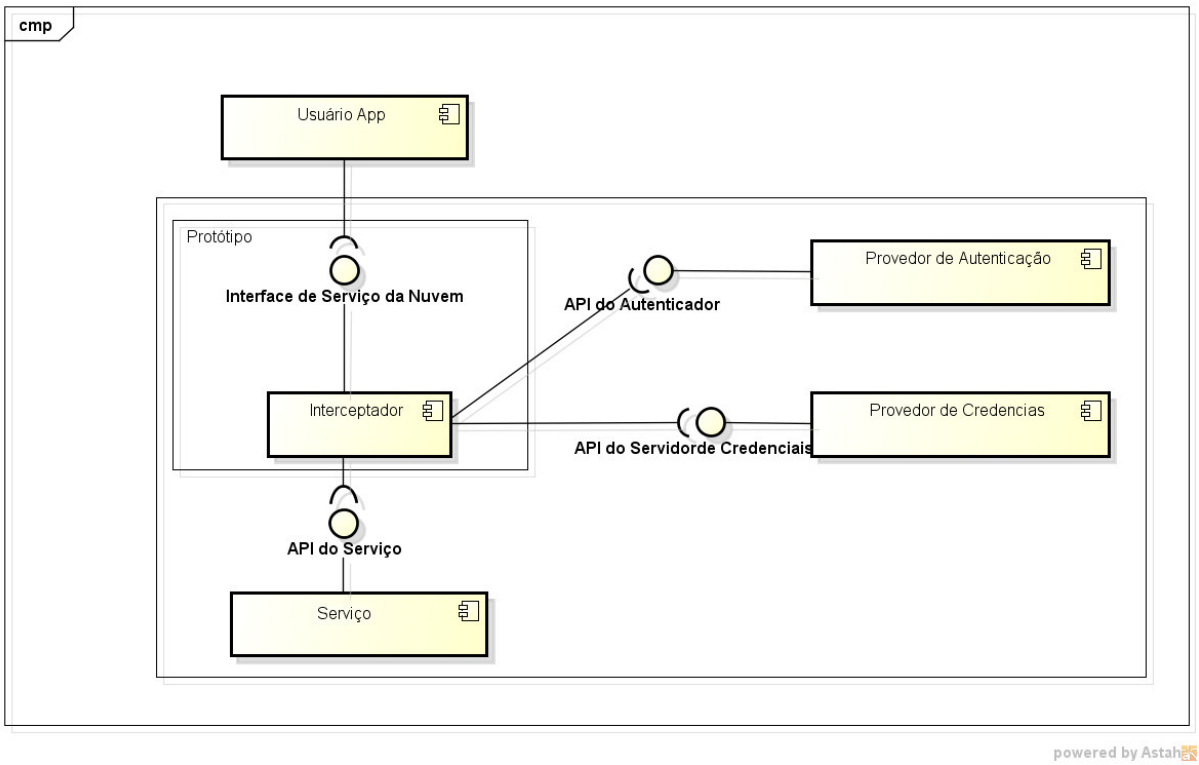

Figura 4.2: Componentes da Arquitetura Desenvolvida

O componente denominado de *Usuário App* inicia a comunicação com o serviço, realizando um solicitação por meio do protocolo HTTP através do navegador web. As solicitações são enviadas para o *Interface de Serviço da Nuvem*, que é a ligação do *Usuário App* ao *Serviço* (a aplicação da nuvem) por ele solicitados.

As solicitações do usuário devem conter credenciais de autenticação, geralmente, usuário e senha, requeridas pelo serviço. Essas credenciais são direcionadas à *Interface de Serviço da Nuvem* e são processadas por um *Interceptador*. Esse *Interceptador* é responsável pela análise e filtragem das credenciais de autenticação do usuário, funções necessárias para a concretização do acesso às aplicações oriundas das plataformas de nuvens.

O *Provedor de Autenticação* usa os atributos de autenticação do usuário para autenticar o acesso às aplicações SaaS. Após validada a autenticidade do *Usuário App*, o *Interceptador* utiliza a *API do Servidor de Credenciais* para obter uma cópia das credenciais de identificação, que poderão ser replicadas para o SP de outra nuvem com o auxílio do protótipo também implantado nesse domínio, caso uma nova requisição de aplicação nela contido venha ser efetuada pelo *Usuário App*.

A credencial de identificação é incorporada à solicitação do serviço, que é em seguida é transferida para a *API do Servidor de Credenciais*, responsável pelo processamento do *Serviço*.

4.4.2 Funcionamento

O processo de autenticação de usuários em sistemas distribuídos envolve, em um primeiro passo, um procedimento de *login* em que um usuário, por meio de trocas com um provedor de identidade (usando técnicas como *userid/password*, *userid/certificate*, *Challenge/Response*, etc.), tem a sua identidade validada diante do sistema. Esse procedimento normalmente termina com o usuário recebendo do seu IdP um *token* ou asserção de autenticação que serão usados para atestar sua autenticidade diante de outras entidades do sistema em subsequentes interações [7].

Nesta pesquisa são utilizados os IdPs (*Shibboleth* e *SimpleSAMLphp*) que, na visão do usuário, são semelhantes e efetuam bem a operação *Single Sign-On (SSO)*. A sequência de passos para os dois IdPs é igual: o SP redireciona o cliente para o IdP, que, autentica o usuário e redireciona o cliente de volta ao SP passando dados que comprovam a autenticação.

O protótipo desenvolvido não altera o fluxo dessas entidades e nem a estrutura o protocolo *SAML* envolvido neste processo de funcionamento. O diagrama

de atividade da Figura 4.3 apresenta o fluxo de autenticação entre os IdPs distribuídos para concretização do SSO em Nuvens para acesso a suas aplicações. Os passos para a realização da integração federada entre os IdPs com o auxílio do protótipo desenvolvido é descrito a seguir.

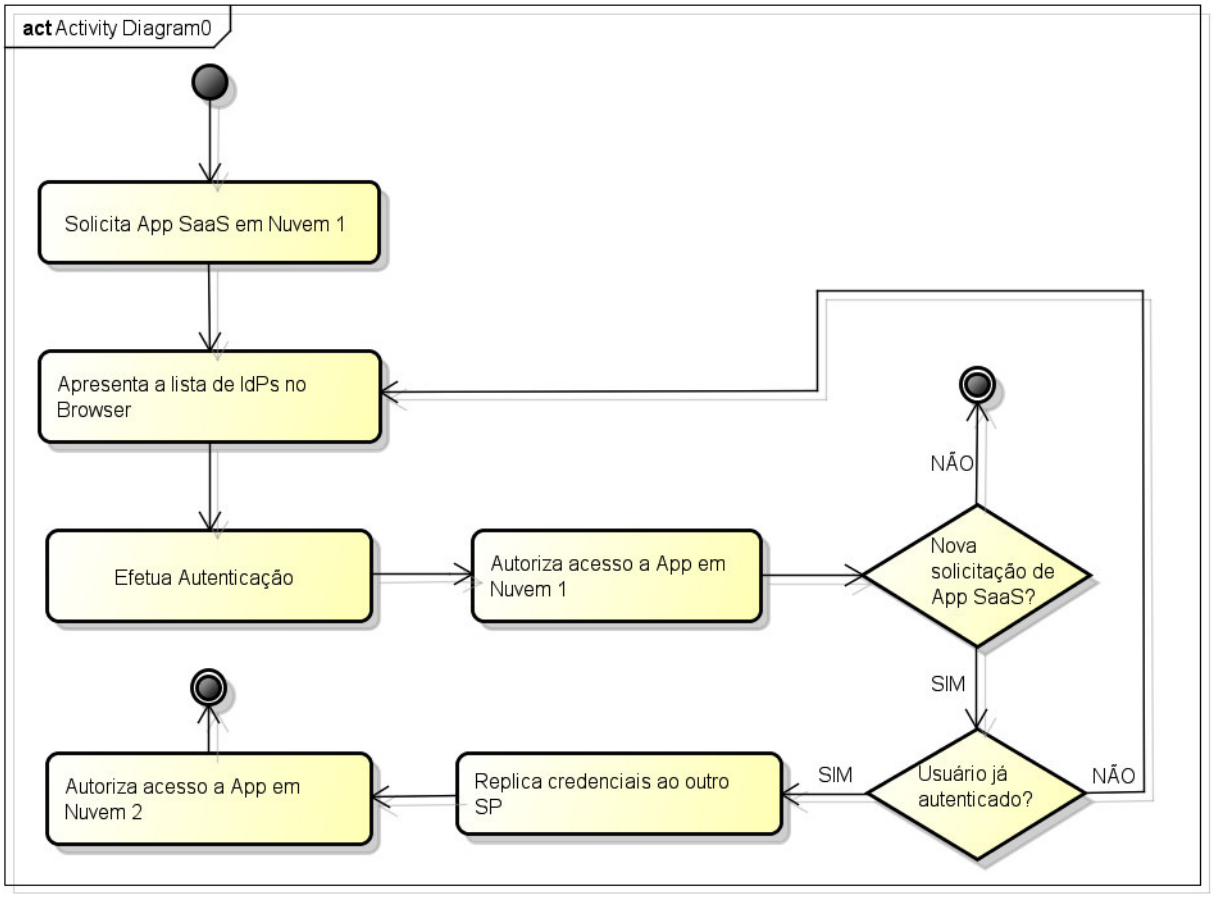

Figura 4.3: Fluxo de Autenticação em Nuvem via SAML

As primeiras 4 atividades (a partir de *Solicita App SaaS em Nuvem 1*) da Figura 4.3 operam de forma similar ao protocolo *SAML* visto na seção 3.4. A interceptação e captura das informações de autenticação do usuário ocorre após a emissão das asserções *SAML* pelo IdP1 encaminhadas ao SP1 da Nuvem 1, local em que acontece o primeiro acesso a uma aplicação, especificamente este processo fica entre entre as atividades *"Efetua Autenticação"* e *"Autoriza acesso a App em Nuvem 1"*.

Após isso, assume-se que o mesmo usuário inicie uma nova requisição de acesso a outra aplicação localizada em Nuvem 2. Nesse ponto, o protótipo verifica se esse usuário já foi autenticado com base nas asserções *SAML* de autenticação anteriormente capturadas. Em caso positivo, o protótipo replica essas credenciais do

usuário para o SP2 referente a segunda nuvem, e por fim, concede o acesso a nova aplicação desejada.

4.5 Testes e Resultados Experimentais

Foram realizados alguns testes que incluíram a infraestrutura de autenticação federada em ambiente de Computação em Nuvem e o protótipo desenvolvido, visando a obtenção de resultados experimentais. A infraestrutura foi implementada majoritariamente sobre um ambiente de máquina virtual, o que viabilizou a fácil portabilidade para plataforma de nuvem.

Os testes realizados tiveram por objetivo principal validar a infraestrutura de autenticação federada em nuvem especialmente a atuação do protótipo na concretização do processo *Single Sign On (SSO)* nos IdPs em nuvens.

Nesse cenário, cada domínio se diferencia em relação sistema de gerenciamento de identidades utilizado (*Shibboleth* e *SimpleSAMLphp*). Analisando os resultados coletados, foi possível notar a integração de identidades federadas para o acesso às aplicações de nuvens baseadas no protocolo *SAML* e o sucesso obtido nas etapas de implementação, implantação em nuvem e fluxo SSO entre os provedores de identidades.

Inicialmente foi necessária implementação dos componentes básicos do cenário, incluindo IdPs, SPs e IdP-Proxys. Especificamente, para instalação do *framework Shibboleth*, contou-se com suporte do Projeto *Gid Lab* (Laboratório de Experimentação em Gestão de Identidade) da Rede Nacional de Pesquisa (RNP) por meio da Federação para Experimentação *Shibboleth* chamada de *"CAFe Expresso"* (*http://wiki.rnp.br/display/gidlab*). E em relação ao *framework SimpleSAMLphp* seguiram-se os procedimentos descritos no site oficial da ferramenta (*https://simplesamlphp.org/*).

Esses componentes foram configurados em máquinas virtuais (conforme Figura 4.4) para que posteriormente pudessem ser migrados para uma plataforma de nuvem. A tecnologia de virtualização utilizada foi KVM *(Kernel-based Virtual Machine)*, sendo que as máquinas virtuais executam um sistema baseado em Linux, como por exemplo, a distribuição *Debian*. E como ambiente de *IaaS*, a plataforma de nuvem utilizada foi a versão de código aberto do software *Eucalyptus*.

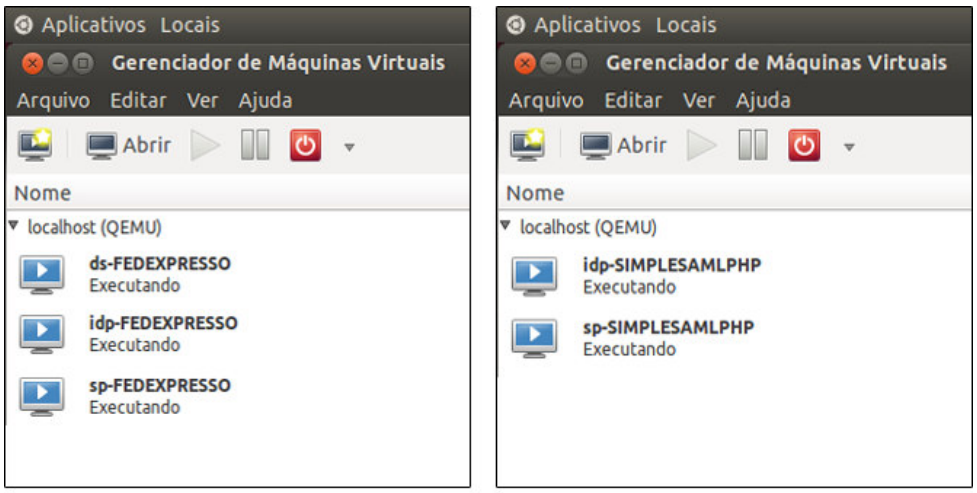

Figura 4.4: Máquinas Virtuais dos *Frameworks* de Autenticação

Para comprovar a aplicabilidade do protótipo desenvolvido, foi criada a seguinte estrutura de sítios, conforme a Tabela 4.2. Contudo, vale ressaltar que não é foco deste trabalho especificar e modelar os serviços ou aplicações da Nuvem, devido à grande diversidade de sistemas (lojas, portais, *e-learning*, *e-commerce*, etc.) que poderiam ser implantados.

Tabela 4.2: Federações de Identidades do Experimento

(Entidade)	**URL**	**Usuário**	**Afiliados**
SP-Shibboleth	sp.fedexpresso.edu.br	-	-
IdP-Shibboleth	idp.fedexpresso.edu.br	aluno	GID LAB, idp-fedlabsac
DS-Shibboleth *(IdP-Proxy)*	ds.fedexpresso.edu.br	-	-
SP-SimpleSAMLphp	sp.fedlabsac.dee.ufma.br	-	GID LAB, idp-fedlabsac
IdP-SimpleSAMLphp	idp.fedlabsac.dee.ufma.br	student	-

O que se busca é estabelecer um conjunto mínimo de requisitos suficientes para permitir a criação ou integração de aplicações com os mecanismos *SAML* para efetuar a SSO em nuvens. Portanto, assume-se que a aplicação de nuvem é solicitada pelo usuário por meio da URL do provedor de serviço de qualquer nuvem, em que páginas Web poderiam ser hospedadas. O SP contém as páginas protegidas, como exemplo tem-se os serviços do *Shibboleth* ofertados em *"sp.fedexpresso.edu.br"*, exigindo a autenticação do usuário.

Para constituir a federação e o círculo de confiança entre nuvens, foi necessário configurar os metadados para cada serviço para estabelecer as relações de

confiança com os IdPs. Da mesma forma, os metadados dos IdPs foram configurados para confiarem nos SPs. Além disso, o protótipo utilizado em cada IdP foi definido com o baseado em senha, porém nenhuma base de dados foi configurada.

Cada provedor de identidade também é responsável pela gestão das relações de confiança com os provedores de serviços. As asserções *SAML* que indicam a autenticação e atributos são digitalmente assinadas pelo provedor de identidade, de modo que a sua validade possa ser verificada pelo provedor de serviço.

As asserções *SAML* emitidas fazem uso de pseudônimos, preservando a privacidade do usuário final. Ou seja, os provedores de serviços não sabem a verdadeira identidade do usuário. No entanto, o provedor de identidade poderia rastrear todos os usuários finais, uma vez que tem de gerar um afirmação cada vez que precisar acessar um provedor de serviços. Além disso, os usuários devem inicializar a transação para permitir que o provedor de serviços obtenha os atributos, tornando o modo *offline* exigência difícil de alcançar.

Tendo como base o fluxo de autenticação federada proposto e visto na Seção 4.4.2, foram realizados 02 testes para verificar a ocorrência do SSO em nuvens. No primeiro, foi testada a operação de autenticação única no sentido da Nuvem1 (contendo o *IdP Shibboleth*) para Nuvem 2 (contendo o *IdP SimpleSAMLphp*) com o intuito de observar o uso de uma mesma crendencial de acesso do usuário a ambos SPs, e no segundo, verificado o mesmo procedimento, o fluxo SSO no sentido inverso, com o objetivo de validar o protótipo usado no auxílio da integração de identidades entre as federações. Estes dois cenários de testes são descritos a seguir.

CENÁRIO DE TESTE 1

Nesse primeiro cenário, considera-se a interação entre as entidades de um sistema de gerenciamento de identidade vista na Seção 2.3.4. O usuário solicita acesso a serviço no SP1 da Nuvem 1 para consumir uma aplicação web via URL *http://sp.fedexpresso.edu.br/*. Esse SP redireciona o navegador do usuário para o IdP1 que necessita dos atributos (conforme Figura 4.5).

Figura 4.5: Tela de Autenticação no IdP Shibboleth

Após identificação do usuário, suas credenciais de acesso são expedidas para o SP para concessão do serviço. Nesse ponto, o protótipo contido na Nuvem 1 intercepta essas credenciais e as replica para a outra nuvem, que também possui uma cópia do protótipo, que controla temporariamente a credencial replicada. Quando no domínio de origem a sessão do usuário é interrompida ou fechada, o protótipo da Nuvem1 anuncia a revogação da credencial compartilhada. Esse procedimento é necessário para aumentar a segurança, uma vez que o usuário só poderá acessar aplicação de outra nuvem, somente enquanto sua sessão estiver ativa.

O resultado obtido por conta da atuação do protótipo é visto na Figura 4.6, que apresenta a validação das mesmas informações de autenticação do usuário (por exemplo, *"aluno"*) logado no IdP1 da Nuvem 1, que foram capturadas e replicadas para o SP2 contido na Nuvem 2, inerente a URL de acesso *http://sp.fedlabsac.dee.ufma.br*.

Figura 4.6: Autenticação com sucesso no SP SimpleSAMLphp

CENÁRIO DE TESTE 2

No sentido contrário do cenário anterior, o segundo cenário simula acesso de um usuário cadastrado no IP2 da Nuvem 2, cujas credenciais de acesso são compartilhadas como auxílio do protótipo para obtenção de acesso a um serviço (SP2) da Nuvem 2. A Figura 4.7 apresenta a tela de autenticação do usuário *"student"* registrado no IdP SimpleSAMLphp (IdP2).

Figura 4.7: Tela de Autenticação no IdP SimpleSAMLphp

Uma vez autenticado, o usuário *"student"* acessa as aplicações da Nuvem 1 e da Nuvem 2, pois suas informações de autenticação são compartilhadas do seu IdP (*http://idp.fedlabsac.dee.ufma.br/*) para o provedor de serviços do *Shibboleth*, conforme se observa na Figura 4.8 que apresenta o resultado da solicitação de acesso a uma aplicação web na Nuvem 1 através da URL *http://sp.fedexpresso.edu.br/*. Garantindo assim a validação da proposta desta pesquisa de efetuar o SSO ou autenticação única entre diferentes provedores de identidades em nuvens.

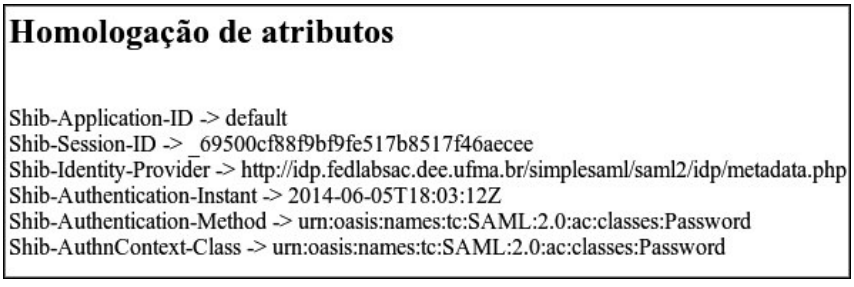

Figura 4.8: Autenticação com sucesso no SP Shibboleth

4.6 Considerações Finais

Este capítulo descreveu como o protótipo desenvolvido foi implementado e testado, e quais resultados experimentais foram obtidos. Através dessa descrição é possível notar que alguns resultados importantes foram verificados, os quais incluem especialmente:

- Portabilidade fácil das infraestruturas virtualizadas de autenticação federada dos *frameworks Shibboleth* e *SimpleSAMLphp* para plataforma de nuvem;
- A utilização do protótipo entre o nível de SaaS e os sistemas de gerenciamento de identidades em nível de PaaS permite a integração de identidades federadas entre nuvens por meio de interceptadores de asserções *SAML*, de forma transparente ao usuário SaaS, dos provedores de serviços e dos provedores de identidades;
- As tecnologias sugeridas são maduras e testadas em produção, o que sugere uma robustez inerente ao modelo proposto.

A metodologia de realização dos testes consistiu em verificar a realização do processo SSO em nuvens compostas por diferentes IdPs, mostrando a atuação do protótipo e o uso do protocolo *SAML* no processo de integração de identidades federadas para acesso a aplicações de nuvens.

5 Conclusões

O desafio da gerência de identidades federadas está relacionado com a problemática de diferentes provedores de serviços utilizarem diferentes mecanismos de mapeamento de usuários em contextos de autenticação [22]. Além disso, o ambiente de Computação em Nuvem distribuído na Internet concentra diversas ferramentas de tecnologias de gerenciamento de identidades, caracterizando-o como complexo e heterogêneo, e despertando a necessidade de pesquisas focadas na privacidade dos dados dos usuários e na interoperabilidade dos sistemas envolvidos.

Este trabalho considera relevante integrar identidades federadas para favorecer o *Single Sign-On (SSO)* ou autenticação única por intermédio de múltiplos domínios, tornando-se um facilitador para o acesso a aplicações de várias nuvens. A integração de identidades é um mecanismo que pode ser utilizado em conjunto com os sistemas de gerenciamento de identidades federadas para promover o compartilhamento dos atributos dos usuários coletados de múltiplos provedores de identidade distribuídos.

O gerenciamento de identidades federadas tem o objetivo de unificar a validação de contexto de usuários nos diferentes domínios parceiros. Isso significa que é preciso manter as devidas restrições e controle de acesso de acordo com o contexto em que o usuário está inserido no momento. As identidades de usuários fornecidas para aplicações podem ser do tipo usuário/senha, certificados digitais, *tokens*, biometria, cartões, entre outros [10].

Dentre os principais desafios da gerência de identidades federadas está o acesso seguro a dados de outros domínios, sem que seja necessária a replicação dos serviços administrativos. Por isso, é necessário que os mesmos protocolos sejam utilizados pelos domínios parceiros, ou que esses protocolos sejam interoperáveis.

O objetivo deste trabalho foi apresentar uma estratégia de integração de identidades federadas entre diferentes provedores de identidade (IdPs) para acesso a aplicações em múltiplas nuvens, garantindo a privacidade, por meio de um modelo baseado em *SAML*, capaz de integrar atributos de usuário assinados digitalmente

pelos próprios IdPs, e mediado em conjunto com as ferramentas de gerenciamento de identidades, utilizadas para o estabelecimento da interoperabilidade entre os provedores de identidade.

Os resultados obtidos possibilitaram a comprovação das hipóteses de pesquisa. Demonstrou-se que o protótipo desenvolvido traz mais flexibilidade aos sistemas de gerenciamento de identidades federadas ao permitir que provedores de serviços possam exigir atributos de múltiplos provedores de identidades localizados em diferentes nuvens.

Comprovou-se ainda nos experimentos que o protótipo desenvolvido em conjunto com os provedores de identidades no ambiente de nuvem, garante a privacidade dos usuários sem prejudicar a interoperabilidade entre os sistemas de gerenciamento de identidades federadas utilizados, e ainda favorece a usabilidade dos usuário, que precisam de uma única autenticação para usufruir das aplicações de várias nuvens.

Sistemas de gerenciamento de identidades federadas podem fornecer a base para a integração de atributos dos usuários em aplicações SaaS de Nuvem. Esses sistemas são sólidos e garantem o acesso federado de seus usuários, porém, muitas vezes, questões referentes à privacidade dos usuários não são devidamente consideradas.

Para contornar o comprometimento das informações do usuário, alguns trabalhos descritos na literatura visam resolver o problema referente à privacidade dos usuários por meio de soluções de gerenciamento de identidades federadas centrado no usuário, uma abordagem que tem a função de atribuir o controle das informações aos próprios usuários.

A vantagem dessa abordagem é a possibilidade do usuário poder escolher o provedor de identidade que deseja utilizar a qualquer momento, ou trocá-lo, e continuar podendo acessar os serviços comumente requeridos, além de utilizar múltiplos provedores de identidade implantadas em outras nuvens.

5.1 Contribuições

A principal contribuição desta pesquisa está no desenvolvimento de um modelo de integração de identidades federadas para viabilizar o acesso a aplicações SaaS controladas por diferentes IdPs implantados em distintas nuvens.

Em geral, esta dissertação diferencia-se em relação trabalhos citados, porque enquanto eles abordam a integração de identidades federadas implementada em uma única nuvem para acesso a aplicações SaaS ou visando estabelecer a ligação de identidades federadas entre camadas de uma única nuvem (IaaS, PaaS e SaaS), este trabalho se concentrou no uso de integração de identidades federadas para possibilitar o acesso a aplicações SaaS em um ambiente multidomínio composto de várias nuvens. Além disso, apresenta experimentalmente a interação, não tratada por outros autores, entre as tecnologias de IdPs (*Shibboleth* e *SimpleSAMLphp*) aliadas no fornecimento de *Single Sign On (SSO)* em nuvem.

Além disso, outras contribuições são identificadas como segue:

- Levantamento bibliográfico abrangente dos principais conceitos de segurança da informação, destacando os padrões de segurança da informação, que contempla um tema atual e relevante para a Governança de TI. Resumidamente, definições, modelos e segurança de Computação em Nuvem, fornecendo uma visão clara deste campo de estudo;
- Descrição dos principais órgãos e iniciativas de normalização de gerenciamento de identidades, apresentando objetivos e características, servindo de apoio na condução de novas pesquisas na área;
- Revisão ampla dos principais conceitos de gerenciamento de identidades e apresentação do estado da arte do tema "Identidades Federadas em Computação em Nuvem" baseadas em publicações recentes;
- Definição de um modelo para integração de identidades federadas. Essa abordagem possibilita o controle de acesso a aplicações SaaS de várias nuvens por meio de uma única autenticação.

Por considerar domínios onde há a predominância do uso do protocolo *SAML*, a abordagem proposta neste trabalho possibilita a integração de vários

IdPs distribuídos em ambiente de Computação em Nuvem, conforme mostram os resultados experimentais na Seção 4.5.

5.2 Limitações

Dentre as propostas de integração de identidades, destacam-se aquelas que fazem uso de interceptadores como o caso desta pesquisa (também conhecida como abordagem baseada em *proxy*). Assim, esta pesquisa identificou as seguintes limitações:

- Essas soluções têm vantagens quanto à sua implementação, porém, é necessário que o modelo proposto contemple a transposição de outros padrões e formatos de credenciais (por exemplo, *WS-Federation*), aumentando a complexidade de desenvolvimento;
- Devido ao uso de uma terceira parte confiável que possa rastrear as interações entre o usuário e os provedores de identidade e de serviço, é necessário investigar melhor a privacidade do usuário;
- Outra desvantagem do uso de interceptadores é que há uma perda de eficiência, uma vez que todas as requisições passam por ele.

5.3 Trabalhos futuros

Este trabalho apresentou um modelo para integração de identidades federadas para acesso a aplicações de nuvens, que, apesar de efetuar a troca de informações de identificação do usuário, possibilitando o *Single Sing On (SSO)* em nuvens, não explora todas as possibilidades que a infraestrutura de autenticação federada apresentada oferece. Dessa forma, esta pesquisa abre um leque para a realização de trabalhos futuros, como os citados a seguir:

- Efetuar uma comparação de diferenças e semelhanças com dois *frameworks* de gerenciamento de identidades utilizados, definindo-se critérios de acoplamento a outros *frameworks* de federação como ao OpenAM, por exemplo;

- Avaliar o modelo proposto e *framewroks* utilizados em relação à facilidade de integração, eficiência, disponibilidade e desempenho. Em relação a este último aspecto, sugere-se o uso da ferramenta *JMeter*;

- Estender o cenário proposto inserindo mais um domínio de nuvem contendo outro framework de federação, por exemplo o OpenAM, ou seja, verificar a escalabilidade da integração e adaptabilidade do protótipo desenvolvido;

- Realizar um análise de segurança da infraestrutura e do modelo propostos;

- Explorar a autorização de acesso entre as plataformas de nuvens por meio da integração de identidades federadas com uso de mecanismo como *ABAC (Attribute Based Access Control)* e do protocolo *eXtensible Access Control Markup Language (XACML)* no *Shibbleth*.

Referências Bibliográficas

[1] *KVM. Kernel-based virtual machine. Site Oficial da Ferramenta*, 2014. Disponível em http://www.linux-kvm.org/page/Main_Page, Acesso em 26-01-2014.

[2] *UNINETT. SimpleSAMLphp. Site Oficial da Ferramenta*, 2014. Disponível em http://simplesamlphp.org, Acesso em 26-01-2014.

[3] AMAZON. Disponível em http://aws.amazon.com/pt/, Acesso em 12-12-2013.

[4] E. Androulaki, B. D. Vo, and S. M. Bellovin. Cybersecurity through an identity management system. 2009.

[5] P. Angin, B. Bhargava, R. Ranchal, N. Singh, M. Linderman, L. Ben Othmane, and L. Lilien. An entity-centric approach for privacy and identity management in cloud computing. In *Reliable Distributed Systems, 2010 29th IEEE Symposium on*, pages 177–183. IEEE, 2010.

[6] Associação Brasileira de Normas Técnicas (ABNT). *NBR ISO/IEC 27002:2013. Tecnologia da informação — Técnicas de segurança — Código de prática para controles de segurança da informação*.

[7] L. Barreto, F. Siqueira, J. Fraga, and E. Feitosa. An intrusion tolerant identity management infrastructure for cloud computing services. pages 155–162, 2013.

[8] E. Bertino and K. Takahashi. *Identity Management: Concepts, Technologies, and Systems*. Artech House, 2010.

[9] A. Bhargav-Spantzel, J. Camenisch, T. Gross, and D. Sommer. User centricity: a taxonomy and open issues. *Journal of Computer Security*, 15(5):493–527, 2007.

[10] Y. Cao and L. Yang. A survey of identity management technology. In *Information Theory and Information Security (ICITIS), 2010 IEEE International Conference on*, pages 287–293. IEEE, 2010.

[11] D. Catteddu. *Cloud Computing: benefits, risks and recommendations for information security*. Springer, 2010.

[12] D. W. Chadwick. Federated identity management. In *Foundations of Security Analysis and Design V*, pages 96–120. Springer, 2009.

[13] D. W. Chadwick, M. Casenove, and K. Siu. My private cloud–granting federated access to cloud resources. *Journal of Cloud Computing*, 2(1):1–16, 2013.

[14] D. W. Chadwick and G. Inman. Attribute aggregation in federated identity management. *Computer*, 42(5):33–40, 2009.

[15] D. W. Chadwick, K. Siu, C. Lee, Y. Fouillat, and D. Germonville. Adding federated identity management to openstack. *Journal of Grid Computing*, 12(1):3–27, 2014.

[16] D. B. Chapman, E. D. Zwicky, and D. Russell. *Building internet firewalls*. O'Reilly & Associates, Inc., 1995.

[17] W. R. Cheswick, S. M. Bellovin, and A. D. Rubin. *Firewalls and Internet security: repelling the wily hacker*. Addison-Wesley Longman Publishing Co., Inc., 2003.

[18] Cloud Security Alliance (CSA). *Cloud Computing Vulnerability Incidents: A Statistical Overview*. Disponível em https://cloudsecurityalliance.org/research/vulnerabilities/#_downloads, Acesso em 12-12-2013.

[19] DAIDALOS (2006). The Daidalos Project. Disponível em hhtp://www.istdaidalos.org, Acesso em 04-12-2013.

[20] S. A. de Chaves, R. B. Uriarte, and C. B. Westphall. Implantando e monitorando uma nuvem privada. In *VIII Workshop em Clouds, Grids e Aplicaç oes*, 2010.

[21] T. El Maliki and J.-M. Seigneur. A survey of user-centric identity management technologies. In *Emerging Security Information, Systems, and Technologies, 2007. SecureWare 2007. The International Conference on*, pages 12–17. IEEE, 2007.

[22] T. El Maliki and J.-M. Seigneur. A survey of user-centric identity management technologies. In *Emerging Security Information, Systems, and Technologies, 2007. SecureWare 2007. The International Conference on*, pages 12–17. IEEE, 2007.

[23] A. J. Feldman, A. Blankstein, M. J. Freedman, and E. W. Felten. Privacy and integrity are possible in the untrusted cloud. *IEEE Data Eng. Bull.*, 35(4):73–82, 2012.

[24] G. Feliciano, L. Agostinho, E. Guimarães, and E. Cardozo. Gerência de identidades federadas em nuvens: Enfoque na utilização de soluções abertas. *Short course, XI SBSeg, Brazil*, 2011.

[25] N. C. Fernandes, M. D. Moreira, P. B. Velloso, L. Costa, and O. Duarte. Ataques e mecanismos de segurança em redes ad hoc. *Minicursos do Simpósio Brasileiro em Segurança da Informaçao e de Sistemas Computacionais (SBSeg'2006)*, pages 49–102, 2006. Disponível em http://www.gta.ufrj.br/ftp/gta/TechReports/FMVC06.pdf, Acessado em 20-09-2013.

[26] D. Ferraiolo, D. R. Kuhn, and R. Chandramouli. *Role-based access control*. Artech House Boston, 2007.

[27] FIDIS (2010). Future of Identity in the Information Society. Disponível em www.fidis.net, Acesso em 04-12-2013.

[28] B. A. Forouzan. *Comunicação de dados e redes de computadores*. Bookman, 4 edition, 2008.

[29] R. Gellman. Privacy in the clouds: risks to privacy and confidentiality from cloud computing. In *Proceedings of the World privacy forum,*, 2012.

[30] Gmail. Disponível em https://mail.google.com, Acesso em 12-12-2013.

[31] GoogleApp. Disponível em http://www.google.com/intx/pt-br/enterprise/apps/business/, Acesso em 12-12-2013.

[32] A. Gopalakrishnan. Cloud computing identity management. *SETLabs briefings*, 7(7):45–54, 2009.

[33] B. Grobauer, T. Walloschek, and E. Stocker. Understanding cloud computing vulnerabilities. *Security & Privacy, IEEE*, 9(2):50–57, 2011.

[34] C. Hare and K. Siyan. Internet firewalls and network security. 1996.

[35] H. Y. Huang, B. Wang, X. X. Liu, and J. M. Xu. Identity federation broker for service cloud. In *Service Sciences (ICSS), 2010 International Conference on*, pages 115–120. IEEE, 2010.

[36] B. Hulsebosch, M. Wegdam, B. Zoetekouw, et al. Virtual collaboration attribute management. 2011.

[37] IBM. Segurança e ampla disponibilidade em ambientes de computação em nuvem. 2011. Disponível em ftp://ftp.software.ibm.com/la/documents/imc/br/news/events/cloud /CloudMSW03010USEN_PTr2.pdf, Acessado em 20-09-2013.

[38] G. Inman and D. Chadwick. A privacy preserving attribute aggregation model for federated identity managements systems. *Upgrade*, 11(1):182–196, 2010.

[39] International Data Corporation (IDC). Disponível em http://www.idc.com/, Acesso em 12-12-2013.

[40] International Data Corporation (IDC). *New IDC IT Cloud Services Survey: Top Benefits and Challenges*. Disponível em http://blogs.idc.com/ie/?p=730, Acesso em 12-12-2013.

[41] ITU-T. *NGN identity management framework. Recommendation Y.2720*, 2009. Disponível em http://www.itu.int/rec/T-REC-Y.2720-200901-I, Acesso em 20-02-2014.

[42] A. Jøsang, J. Fabre, B. Hay, J. Dalziel, and S. Pope. Trust requirements in identity management. In *Proceedings of the 2005 Australasian workshop on Grid computing and e-research-Volume 44*, pages 99–108. Australian Computer Society, Inc., 2005.

[43] A. Jøsang and S. Pope. User centric identity management. In *AusCERT Asia Pacific Information Technology Security Conference*, page 77. Citeseer, 2005.

[44] C. Kalloniatis, V. Manousakis, H. Mouratidis, and S. Gritzalis. Migrating into the cloud: Identifying the major security and privacy concerns. In *Collaborative, Trusted and Privacy-Aware e/m-Services*, pages 73–87. Springer, 2013.

[45] Kantara Iniciative. Disponível em http://kantarainitiative.org/, Acesso em 12-12-2013.

[46] N. Klingenstein. Attribute aggregation and federated identity. In *Applications and the Internet Workshops, 2007. SAINT Workshops 2007. International Symposium on*, pages 26–26. IEEE, 2007.

[47] U. Lampe, O. Wenge, A. Müller, and R. Schaarschmidt. On the relevance of security risks for cloud adoption in the financial industry. 2013.

[48] C. E. Landwehr. Computer security. *International Journal of Information Security*, 1, 2003.

[49] M. A. Leandro, T. J. Nascimento, D. R. dos Santos, C. M. Westphall, and C. B. Westphall. Multi-tenancy authorization system with federated identity for cloud-based environments using shibboleth. In *ICN 2012, The Eleventh International Conference on Networks*, pages 88–93, 2012.

[50] F. Lombardi and R. Di Pietro. Secure virtualization for cloud computing. *Journal of Network and Computer Applications*, 34(4):1113–1122, 2011.

[51] A. M. Lonea, H. Tianfield, and D. E. Popescu. Identity management for cloud computing. In *New Concepts and Applications in Soft Computing*, pages 175–199. Springer, 2013.

[52] F. P. Marzullo. *SOA na Prática: inovando seu negócio por meio de soluções orientadas a serviços*. NOVATEC, 2009.

[53] U. Maurer. Information security (part i). 2013. Disponível em http://people.ee.ethz.ch/ lamy/pdfs/infsec2013-lecture-notes.pdf, Acesso em 22-12-2013.

[54] P. M. Mell and T. Grance. Sp 800-145. the nist definition of cloud computing. Technical report, Gaithersburg, MD, United States, 2011.

[55] R. T. Michael T. Goodrich. *Introdução à Segurança de Computadores*. Bookman, 2013.

[56] R. Morgan, S. Cantor, S. Carmody, W. Hoehn, and K. Klingenstein. Federated security: The shibboleth approach. *Educause Quarterly*, 27(4):12–17, 2004.

[57] M. A. E. Nagano and R. K. Yokoo. Gestão de segurança: proteção da informação e do patrimônio empresarial. 2013.

[58] E. T. NAKAMURA. Geus, paulo lício de. *Segurança de redes em ambientes cooperativos*, 2007.

[59] V. Nicomette. *La protection dans les systèmes à objets répartis*. PhD thesis, Institut National Polytechnique de Toulouse-INPT, 1996.

[60] T. H. Noor, Q. Z. Sheng, S. Zeadally, and J. Yu. Trust management of services in cloud environments: Obstacles and solutions. *ACM Computing Surveys (CSUR)*, 46(1):12, 2013.

[61] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov. Eucalyptus: A technical report on an elastic utility computing architecture linking your programs to useful systems. In *UCSB TECHNICAL REPORT*. Citeseer, 2008.

[62] OASIS. *SAML*, 2013. Disponível em https://www.oasis-open.org/committees, Acesso em 30-11-2013.

[63] OAuth 2.0. Relatório técnico, IETF. Disponível em http://tools.ietf.org/html/draft-ietf-oauth-v2-31, Acesso em 12-12-2013.

[64] OpenID Foundation. Disponível em http://openid.net/, Acesso em 12-12-2013.

[65] S. Pearson and G. Yee. *Privacy and security for cloud computing*. Springer, 2013.

[66] PRIME (2010). Privacy and Identity Management for Europe. Disponível em www.prime-project.eu, Acesso em 04-12-2013.

[67] I. Ray and I. Ray. Trust-based access control for secure cloud computing. In *High Performance Cloud Auditing and Applications*, pages 189–213. Springer, 2014.

[68] A. Reed, C. Rezek, and P. Simmonds. Security guidance for critical areas of focus in cloud computing v3.0. *Cloud Security Alliance*, 2011.

[69] A. G. Revar and M. D. Bhavsar. Securing user authentication using single sign-on in cloud computing. In *Engineering (NUiCONE), 2011 Nirma University International Conference on*, pages 1–4. IEEE, 2011.

[70] M. G. Rita de Castro, Luiza Domingos. Gestão de vulnerabilidades em cloud computing: Um cenário da nuvem pública. 2012. Disponível em http://www.infobrasil.inf.br/userfiles/16-S1-2-97170-GestAcesso em12-12-2013.

[71] C. Rong, S. T. Nguyen, and M. G. Jaatun. Beyond lightning: A survey on security challenges in cloud computing. *Computers & Electrical Engineering*, 39(1):47–54, 2013.

[72] S. Ruj, M. Stojmenovic, and A. Nayak. Privacy preserving access control with authentication for securing data in clouds. In *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*, pages 556–563. IEEE, 2012.

[73] A. Saldhana, A. Nadalin, and M. Rutkowski. Identity in the cloud use cases version 1.0, 2012.

[74] A. C. Sarma and J. Girão. Identities in the future internet of things. *Wireless personal communications*, 49(3):353–363, 2009.

[75] B. Schneir. Applied cryptography. *John Wiley & Sons*, 1996.

[76] I. S. SetteI and C. A. FerrazI. Integrando plataformas de nuvens a federações de identidade. *Anais do 32º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos. SBRC 2014*, 2014.

[77] B. Sosinsky. *Cloud Computing Bible*. Wiley Publishing, 1st edition, 2011.

[78] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. Foster. Virtual infrastructure management in private and hybrid clouds. *Internet Computing, IEEE*, 13(5):14–22, 2009.

[79] W. Stallings. *Cryptography and Network Security: Principles and Practice*. Pearson Education, 2013.

[80] D. Stiawan, A. I. Shakhatreh, M. Idris, K. Abu Bakar, A. H. Abdullah, et al. Intrusion prevention system: a survey. *Journal of Theoretical and Applied Information Technology*, 40(1):44–54, 2012.

[81] M. Stihler, A. O. Santin, A. Marcon, and J. da Silva Fraga. Integral federated identity management for cloud computing. In *New Technologies, Mobility and Security (NTMS), 2012 5th International Conference on*, pages 1–5. IEEE, 2012.

[82] S. Subashini and V. Kavitha. A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, 34(1):1–11, 2011.

[83] H. Susanto12, M. N. Almunawar, and Y. C. Tuan. Information security management system standards: A comparative study of the big five. *International Journal of Electrical & Computer Sciences*, 11(05), 2011.

[84] H. Takabi, J. B. Joshi, and G.-J. Ahn. Security and privacy challenges in cloud computing environments. *Security & Privacy, IEEE*, 8(6):24–31, 2010.

[85] A. S. Tanenbaum. *Sistemas operacionais modernos*. Prentice- Hall, São Paulo, 3 edition, 2010.

[86] A. Tassanaviboon and G. Gong. Oauth and abe based authorization in semi-trusted cloud computing: aauth. In *Proceedings of the second international workshop on Data intensive computing in the clouds*, pages 41–50. ACM, 2011.

[87] R. A. Thomás Filipe da S. Diniz, Carlos Eduardo da Silva. Integração de clientes da nuvem openstack swift com uma federação de identidade. *In: XIII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais. SBC*, pages 455–464, 2013.

[88] G. D. Tormo, F. G. Mármol, and G. M. Pérez. Identity management in cloud systems. In *Security, Privacy and Trust in Cloud Systems*, pages 177–210. Springer, 2014.

[89] J. Torres, M. Nogueira, and G. Pujolle. A survey on identity management for the future network. *Communications Surveys & Tutorials, IEEE*, 15(2):787–802, 2013.

[90] L. M. Vaquero, L. Rodero-Merino, and D. Morán. Locking the sky: a survey on iaas cloud security. *Computing*, 91(1):93–118, 2011.

[91] T. Velte, A. Velte, and R. Elsenpeter. *Cloud computing, a practical approach*. McGraw-Hill, Inc., 2009.

[92] M. Veras. *Arquitetura de Nuvem - Amazon Web Services (AWS)*, volume 1293.6. 1 edition, 2013.

[93] C. Wang, B. Zhang, K. Ren, and J. Wang. Privacy-assured outsourcing of image reconstruction service in cloud. 2013.

[94] P. Yadav, P. Mishra, T. Sharma, and V. Sharma. Security issues in cloud computing and associated mitigation techniques. *International Journal of Innovative Research and Development*, 2013.

[95] R. Yogamangalam and V. S. Sriram. A review on security issues in cloud computing. *Journal of Artificial Intelligence*, 6, 2012.

[96] M. Y. A. Younis and K. Kifayat. Secure cloud computing for critical infrastructure: A survey. *Liverpool John Moores University, United Kingdom, Tech. Rep*, 2013.

[97] D. M. Yuri Diogenes. *Certificação Security+ — Da Prática Para o Exame SY0-301*. Number 9788561893194. 2 edition, 2013.

[98] Q. Zhang, L. Cheng, and R. Boutaba. Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1(1):7–18, 2010.