RONY ROGÉRIO MARTINS SAKURAGUI

GERENCIAMENTO DE IDENTIDADES COM PRIVACIDADE DO USUÁRIO EM AMBIENTE WEB

Tese apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do título de Doutor em Engenharia

São Paulo
2012

RONY ROGÉRIO MARTINS SAKURAGUI

GERENCIAMENTO DE IDENTIDADES COM PRIVACIDADE DO USUÁRIO EM AMBIENTE WEB

Tese apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do título de Doutor em Engenharia

Área de Concentração:
Sistemas Digitais

Orientadora: Profª. Dra. Tereza Cristina Melo de Brito Carvalho

São Paulo
2012

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 10 de janeiro de 2012.

Assinatura do autor _____

Assinatura do orientador _____

FICHA CATALOGRÁFICA

Sakuragui, Rony Rogério Martins
Gerenciamento de identidades com privacidade do usuário em ambiente Web / R.R.M. Sakuragui. -- ed.rev. -- São Paulo, 2012.
135 p.

Tese (Doutorado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1. Internet 2. Protocolos de comunicação 3. Segurança de redes I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II. t.

AGRADECIMENTOS

Em primeiro lugar, o agradecimento à minha orientadora Profª Tereza Cristina Carvalho. Pela sua paciência, suporte e orientação neste trabalho, como também em outros aspectos de minha vida profissional e pessoal.

À minha querida esposa Mônica, pelo seu apoio e encorajamento incondicionais ao longo dos altos e baixos no desenvolvimento deste trabalho.

Aos meus pais, por seus esforços investidos na educação e bem-estar de seu filho. Ao meu irmão André Alves, por seu infalível apoio como "irmão mais velho".

Ao Prof. Marcos Simplício, pelo seu companheirismo e pelas "infinitas" discussões acerca desta tese nas voltas de carro para nossas casas após o expediente do LARC. Para os que conhecem o trânsito na saída da USP em certos horários, certamente sabem o que "infinitas" pode significar neste caso.

À Cristina Dominicini, por sua inestimável ajuda ao longo da elaboração desta tese. Certamente este trabalho não estaria completo sem suas observações e contribuições.

À Ericsson Telecomunicações do Brasil e Ericsson Research da Suécia, pelo suporte financeiro e cooperação conjunta nos projetos de pesquisa do LARC.

Em especial, ao Prof. Wilson Ruggiero, pelas conversas e todo o incentivo durante o desenvolvimento deste trabalho no LARC.

Enfim, a todos os meus amigos, colegas e professores do LARC, que de alguma forma têm suas contribuições no desenvolvimento deste trabalho. O companheirismo dos almoços, cafés e jogos certamente possuem um lugar no desenvolvimento desta tese.

RESUMO

Sistemas de Gerenciamento de Identidade Centrados no Usuário têm sido utilizados na Internet como meio de evitar o gerenciamento de múltiplas contas em sites e serviços na Web. Embora o uso de tais sistemas apresente benefícios, usuários podem ter sua privacidade prejudicada, uma vez que suas identidades tendem a ser conhecidas e controladas por uma entidade central. Dessa maneira, os acessos a serviços e o comportamento dos usuários tendem a ser facilmente rastreáveis em toda a rede. Por outro lado, do ponto de vista dos serviços, existem casos onde o conhecimento e a comprovação de informações do usuário é uma necessidade para o controle de acesso e provimento do serviço. Assim, o objetivo deste trabalho é propor uma solução de gerenciamento de identidades que proteja a privacidade e, ao mesmo tempo, possibilite a comprovação de atributos de identidade do usuário para um provedor de serviços no ambiente Web atual. Esta proposta inova dentre os trabalhos relacionados encontrados na literatura devido à sua adequação às necessidades e limitações existentes no ambiente típico da interação entre usuários e sites na Internet. A verificação do cumprimento dos objetivos de autenticação de atributos de identidade e privacidade do usuário é realizada por meio da análise formal do protocolo da solução. Ainda, com a aplicação de uma métrica, são analisados as condições e níveis de anonimato de um usuário no uso do sistema.

ABSTRACT

User-centric Identity Management Systems have been used on the Internet for avoiding the management of multiple users' accounts in different sites and services on the Web. Although those systems can bring some benefits for its users, their privacy may be jeopardized since their identities are likely to be known and controlled by a central entity. This way, users' behavior and their accesses to services are likely to be easily tracked on the network. On the other side, from the service's point of view, there are occasions where the knowledge and verification of some user's aspects and attributes are necessary for access control and service providing. Thereby, the goal of this work is to propose a solution for identity management that provides enhanced privacy for user and, at the same time, allows them to prove attributes of their identity to a service provider on the current Web environment on the Internet. This proposal innovates when compared to related works due to its suitability to the environment and its interactions between clients and sites on the Internet. The objectives related to the verification of identity's attributes and privacy concerns in this proposal are analyzed by formal methods. This work also presents an analysis on the conditions and levels of anonymity when users interact with the system based on a metric.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AX	*Attribute Exchange*
CP	*Credential Provider*
HTML	*HyperText Markup Language*
HTTP	*HyperText Transfer Protocol*
HTTPS	*HyperText Transfer Protocol Secure*
ICP	*Infra-estrutura de Chaves Públicas*
IdP	*Identity Provider*
IP	*Internet Protocol*
MAC	*Message Authentication Code*
OP	*OpenID Provider*
OSI	*Open Systems Interconnection*
RP	*Relying Party*
SGI	Sistema de Gerenciamento de Identidades
SIM	*Subscriber Identity Module*
TCP	*Transmission Control Protocol*
URL	*Uniform Resource Locator*

SUMÁRIO

1 INTRODUÇÃO

Desde a popularização da Internet, diversos padrões e tecnologias têm surgido e alterado a forma como usuários interagem e acessam serviços nesse ambiente (GARRETT; OTHERS, 2005), (HAMMER-LAHAV; RECORDON, 2010), (CADENHEAD *et al.*, 2006). A partir da Web 2.0 (O REILLY, 2007), o ambiente Web passou a ser visto como uma plataforma de aplicações, ao invés de, apenas, um conjunto de documentos em hipertexto distribuídos em servidores na Internet (MADDEN; FOX, 2007). Dentre as principais mudanças de visão nesse ambiente, um maior destaque tem sido dado à necessidade de aplicações e tecnologias centradas no usuário final (VOSSEN; HAGEMANN, 2007), (MADDEN; FOX, 2007).

Seguindo essa tendência, soluções de gerenciamento de identidade centrados no usuário têm sido propostas para Internet. Tais sistemas permitem que usuários gerenciem suas identidades *on-line* de maneira centralizada e as reutilizem em diferentes contextos, de acordo com suas preferências. Em outras palavras, esses sistemas permitem que usuários possam, por meio de uma **única conta** criada em um provedor de identidade, efetuar *login* em *sites* distintos na Internet[1]. Ainda, provedores de identidade tipicamente permitem o **compartilhamento de dados** pessoais, mediante o consentimento do usuário, com *sites* ou serviços Web no momento de *login*.

Embora tais sistemas de gerenciamento de identidade apresentem diversos benefícios, o seu uso pode causar graves prejuízos à privacidade dos usuários quando estes são associados às suas identidades reais. Diversos provedores de identidade defendem a ideia que usuários devem se identificar e informar seus atributos reais de identidade no momento de registro. Particularmente, Google e Facebook proíbem o uso de nomes

[1] Sites com suporte a *login* por meio de provedores de identidade apresentam uma lista com os provedores mais populares em sua página de *login* (tipicamente na forma de botões com os logotipos de cada provedor). O provedor de identidade é a entidade central de um sistema de gerenciamento de identidades e é responsável pelas asserções a respeito da identidade de seus usuários. Como exemplos de *provedores de identidade* podem-se citar: Google, Facebook e Yahoo. Em 2010, o Yahoo passou também a permitir que usuários acessem seu sistema com o uso de contas fornecidas pelo Google ou Facebook (WU, 2010).

falsos, fictícios ou pseudônimos nos perfis de seus usuários[2] (MCCRACKEN, 2011), (NBC, 2009).

Assim, sem a existência de mecanismos de proteção à privacidade dos usuários em sistemas de gerenciamento de identidade, existe o risco de que tais sistemas se tornem também sistemas de vigilância, capazes de conhecer as identidades reais de seus usuários, rastrear suas atividades e controlar seus dados pessoais.

Por outro lado, do ponto de vista dos serviços, é importante perceber que em alguns casos existe a necessidade real de comprovação de informações do usuário. Por exemplo, *sites* de jogos eletrônicos (e.g., www.steam.com) possuem como requisição legal que seus usuários sejam maiores de 13 anos para o acesso a alguns jogos. Ainda, *sites* de artigos periódicos (e.g.: http://ieeexplore.ieee.org) exigem a comprovação de afiliação a uma determinada organização (e.g.: uma universidade) para permitir o acesso ao seu conteúdo. De modo geral, informações como idade, país, filiação e outras podem ser úteis ou mesmo necessárias para o provimento do serviço a um usuário, ainda que este seja anônimo no sistema.

Com essa discussão, é possível afirmar que existe a necessidade de um sistema ou mecanismo na Web que possa lidar ao mesmo tempo com as questões apontadas aqui sobre **gerenciamento de identidades**, **privacidade** e **comprovação de atributos de usuários**.

Dessa maneira, este trabalho concentra-se em investigar e responder à seguinte pergunta:

> *Como prover um sistema de **gerenciamento de identidades** que proteja a **privacidade** e, ao mesmo tempo, possibilite a **comprovação de atributos de identidade** do usuário para um provedor de serviços na **Web atual**?*

Como resposta a essa questão, este trabalho define e propõe o protocolo e a arquitetura NibbleID. Essa arquitetura é proposta considerando-se as características do ambiente

[2] Em junho de 2011, ocorreu um cancelamento maciço de contas cujos nomes foram detectados como falsos no Google+ (FERNANDES, 2011). A polêmica sobre a proibição do uso de pseudônimos é conhecida como “nymwar” na Internet atualmente (FENTON, 2011).

Web e as possíveis relações de confiança existentes entre usuários e entidades na Internet. Já o protocolo permite que um usuário selecione um conjunto de atributos de sua identidade (identidade parcial) e comprove-os para um provedor de serviços ou *site* na Web por meio de um pseudônimo. Dentro de certas condições, entre elas a escolha adequada de uma identidade parcial, as propriedades do protocolo podem permitir que um usuário, ao acessar um serviço, permaneça anônimo e não rastreável pelas entidades do sistema.

1.1 Background e Motivação

Os estudos sobre gerenciamento de identidades na Internet iniciaram-se pelo autor em um projeto desenvolvido em conjunto com a Ericsson Labs[3] e o Laboratório de Arquitetura e Redes de Computadores (LARC) do departamento de Engenharia de Computação e Sistemas Digitais (PCS) desta universidade. Resumidamente, nesse projeto foi desenvolvido um sistema de gerenciamento de identidade baseado no protocolo OpenID (RECORDON; FITZPATRICK, 2007) que permite que um usuário seja autenticado em diferentes *sites* na Internet por meio do cartão *SIM* em seu telefone celular[4].

Durante as atividades de pesquisa desse projeto, percebeu-se que pouco foi proposto na literatura para a proteção da privacidade de usuários em sistemas de gerenciamento de identidade no ambiente da Web. Embora alguns protocolos para o gerenciamento de identidade com suporte à privacidade tenham sido propostos para ambientes genéricos de rede (como será visto no capítulo 5), nenhuma das propostas encontradas enfoca as necessidades e limitações existentes no ambiente típico da interação entre usuários e *sites* na Web (vistos no capítulo 4).

[3] O Ericsson Labs é o setor da Ericsson Research que suporta pesquisa com tecnologias e soluções inovadoras. O seu portal na Internet encontra-se em https://labs.ericsson.com/

[4] Os resultados podem ser vistos em https://labs.ericsson.com/apis/identity-management-framework/. Esse projeto não abordou questões relacionadas à privacidade do usuário.

Se por um lado existem evidências de que alguns usuários e serviços não se preocupam com a questão da privacidade na Web, principalmente nas redes sociais (BARNES, 2006), por outro lado, diversos protestos e discussões têm surgido devido à crescente diminuição da privacidade de usuários por causa do uso de serviços centralizadores de dados, tais como sistemas de gerenciamento de identidade (FENTON, 2011), (MCCRACKEN, 2011), (NBC, 2009).

Com isso, este trabalho motiva-se em criar uma solução que permita que usuários escolham o nível de privacidade mais adequado a suas necessidades e, ao mesmo tempo, desfrutem dos benefícios proporcionados pelas soluções de gerenciamento de identidades na Web.

1.2 Objetivo e Escopo

Este trabalho tem como objetivo definir uma **arquitetura e um protocolo para um sistema de gerenciamento de identidade com suporte à privacidade do usuário no uso de serviços na Web**. Além disso, essa solução deve permitir que usuários **selecionem e comprovem atributos de suas identidades** para provedores de serviço na Web sem a exposição de informações não consentidas pelo usuário.

O escopo da solução limita-se às **questões de privacidade** do usuário **relacionadas** ao **gerenciamento de identidades** na camada de aplicação. Assim, os mecanismos de suporte à privacidade propostos nesse trabalho não pretendem abordar, por exemplo, as questões relativas às camadas de transporte ou rede no sistema. Para essas camadas, mecanismos de anonimização por meio de redes *overlay*, como o sistema *Tor* (DINGLEDINE; MATHEWSON; SYVERSON, P., 2004), *Anonymizer* (BOYAN, 1997) ou o trabalho proposto em (REITER; RUBIN, 1998) podem ser utilizados em conjunto com a solução proposta.

1.3 Organização do texto

O capítulo 2 deste documento apresenta uma visão geral dos sistemas de gerenciamento de identidade. Por sua vez, no capítulo 3 são definidos os termos relacionados a privacidade, anonimato e credenciais usados neste trabalho. Então, no capítulo 4, é apresentada uma descrição do problema e dos desafios do ambiente Web, assim como os requisitos de uma solução para este ambiente. Em seguida, o capítulo 5 agrupa e apresenta os trabalhos relacionados, apontando suas deficiências no atendimento aos requisitos descritos no capítulo anterior. Com isso, o capítulo 6 descreve a arquitetura e o protocolo NibbleID, frutos dos objetivos deste trabalho. Na sequencia, uma análise formal do protocolo é vista no capítulo 7. No capítulo 8, é apresentada uma análise do grau de anonimato provido pela solução. Por fim, o capítulo 9 apresenta as considerações finais deste trabalho.

2 SISTEMAS DE GERENCIAMENTO DE IDENTIDADE

"I am absolutely opposed to a national ID card. This is a total contradiction of what a free society is all about. The purpose of government is to protect the secrecy and the privacy of all individuals"
Ron Paul- Congressista norte-americano

Este capítulo apresenta os conceitos de **identidade** e **sistemas de gerenciamento de identidade** de acordo com a literatura especializada na área. Ainda, este capítulo relaciona as tecnologias mais relevantes para o gerenciamento de identidades, destacando o OpenID como um padrão emergente na Internet.

A compreensão do estado atual dos sistemas de gerenciamento de identidade na Web, em particular o OpenID, é vital para a compreensão da proposta deste trabalho.

É importante explicitar ainda que, de modo geral, os sistemas de gerenciamento de identidade definidos e abordados neste capítulo não apresentam em sua concepção mecanismos de preservação de privacidade do usuário.

2.1 Identidade

Identidade pode ser compreendida como a representação de uma **entidade** em um domínio por meio de um conjunto de **atributos** (JOSANG; POPE, 2005), (PFITZMANN; HANSEN, 2005), (MIYATA *et al.*, 2006). Uma **entidade** pode ser entendida como uma pessoa ou organização do domínio. Já um **atributo** representa uma característica da entidade dentro de um domínio particular e pode ou não ser única dentro desse domínio (JOSANG; POPE, 2005), (BHARGAV-SPANTZEL *et al.*, 2007).

Como exemplo, considere um indivíduo (entidade) como um usuário de um *site* na Internet. O nome de usuário (*username*) desse indivíduo, bem como os dados cadastrais (atributos) apresentados no momento de registro, representam a identidade do usuário dentro do domínio do *site*. Essa mesma identidade possivelmente não será válida em outro *site* ou serviço na Internet (outro domínio).

A Figura 1 a mostra a relação entre entidades, identidades e atributos em um domínio. Note que uma entidade pode possuir mais de uma identidade, dependendo da política exercida pelo domínio.

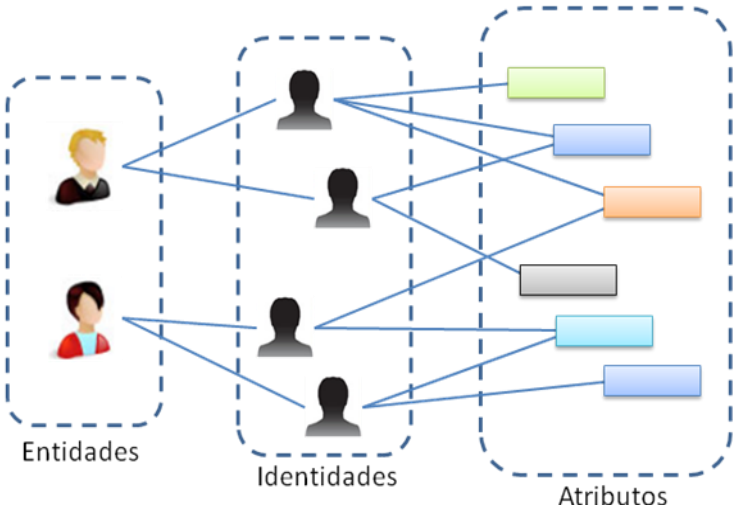

Figura 1 – Correspondência entre entidades, identidades e atributos. Adaptado de (JOSANG; POPE, 2005)

2.2 Sistemas de Gerenciamento de Identidade (SGI)

De acordo com (BHARGAV-SPANTZEL *et al.*, 2007), um sistema de gerenciamento de identidade (SGI) pode ser definido como um conjunto de componentes de *software* e protocolos que manipulam identidades de acordo com um ciclo, chamado de ciclo de identidade. Esse ciclo é composto basicamente por registro, armazenamento, recuperação, fornecimento e revogação de identidades.

Na Internet, com o aparecimento de serviços sendo oferecidos na Web e a necessidade da identificação de usuários, tornou-se também necessário o gerenciamento de identidades de usuários. Atualmente, a forma mais comum de gerenciamento de identidades de usuários na Web é o modelo onde cada *site* ou serviço gerencia, de forma isolada, as contas e informações dos usuários registrados. De modo geral, esse modelo supõe o registro e criação de uma conta de usuário em cada serviço. Tipicamente a criação de conta envolve a criação de um *username* e uma senha que devem ser utilizados pelo usuário em acessos posteriores ao serviço. A Figura 2, a seguir, ilustra

esse modelo onde um usuário mantém diversas contas, uma para cada *site* onde possui registro.

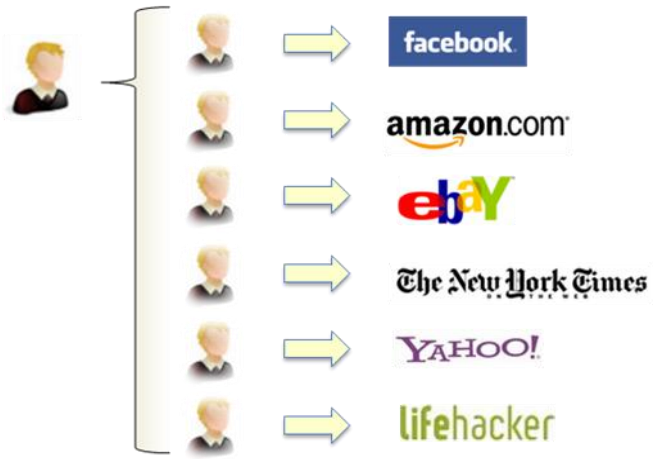

Figura 2 - Gerenciamento de contas múltiplas mantidas por um mesmo usuário

De acordo com (JOSANG; POPE, 2005), esse modelo de SGI pode ser caracterizado como "centrado no provedor de serviço" (*Service Provider Centric*). Em SGIs centrados no provedor de serviço, as identidades dos usuários são controladas pelo *site*/organização. Aos usuários cabe, apenas, a comprovação da posse da identidade por meio de algum mecanismo de autenticação (por exemplo, *username* e senha). A Figura 3, a seguir, ilustra esse modelo de SGI centrado no provedor de serviço, onde o *site* é o detentor das identidades dos seus usuários.

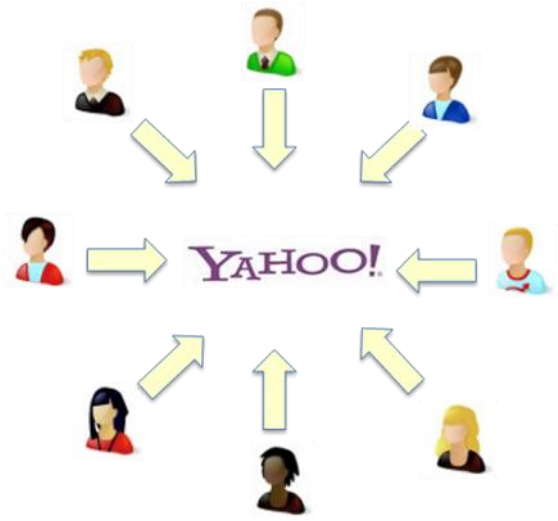

Figura 3 - Modelo de SGI centrado no provedor de serviço

Este modelo, embora ainda predominante na Internet, traz consigo o problema do gerenciamento de múltiplas contas por parte do usuário. Como mencionado anteriormente, em SGIs centrados no provedor de serviço, cada usuário é obrigado a criar e manter diversas contas, uma para cada *site* ou serviço, resultando em múltiplas contas. Um dos problemas típicos dessa abordagem é que o usuário nem sempre consegue lembrar-se das informações de *login* (*username* e senha) para cada serviço em que se registrou. Na maioria dos casos, devido a esse problema, grande parte dos usuários na Internet reutiliza um mesmo *username* e senha no registro em diferentes *sites* (IVES; WALSH; SCHNEIDER, 2004). Isso traz um grave risco de segurança, uma vez que um eventual vazamento das informações de *login* de um *site*, ou mesmo o uso indevido dessas informações, pode, potencialmente, comprometer não apenas a conta em questão, mas também as demais contas daqueles usuários na Internet.

De modo a atenuar esse problema, soluções de gerenciamento de identidade conhecidas como "centradas no usuário" (*User Centric*) (JOSANG; POPE, 2005) têm sido utilizadas na Internet para gerenciamento de contas múltiplas. Por meio do uso de tais sistemas na Internet, torna-se possível para um usuário a criação de uma conta em um serviço de identidades, de modo que a esta mesma conta pode ser reutilizada de forma segura para o *login* em diversos outros serviços ou *sites* na Internet. A Figura 4, a seguir, ilustra um exemplo de serviço na Web que permite o acesso por meio da utilização de contas existentes em outros provedores (no caso, Google, Yahoo, WindowsLiveID, etc.).

Figura 4 - Acesso a um serviço na Web (*http://sonora.terra.com.br*) com a opção de utilização de contas existentes em outros provedores

Em SGIs “centrados no usuário”, o controle das identidades é feito pelo próprio usuário de maneira centralizada e *on-line* (JOSANG; POPE, 2005). As identidades podem ser armazenadas de modo centralizado por organizações independentes[5] ou armazenadas pelos próprios usuários. Em tais SGIs, é possível que um usuário possua mais de uma identidade armazenada e escolha qual deve ser utilizada de acordo com o contexto (VACCA, 2009).

De maneira simplificada, a autenticação em SGIs centrados no usuário ocorre por meio de uma mensagem gerada por um provedor de identidades. Essa mensagem comprova que o usuário possui uma conta válida naquele provedor. Assim, essa mensagem é entregue ao *site* ou serviço na Web, o qual deve ser capaz de verificar a validade de tal mensagem. A tecnologia de SGI centrado no usuário mais utilizada na Internet é o OpenID, a ser discutido adiante na seção 2.4.1.

A Figura 5, a seguir, ilustra esse modelo de SGI centrado no usuário, onde a identidade é controlada pelo próprio usuário.

Figura 5 - Modelo de SGI centrado no usuário

Uma variante híbrida para os SGIs são os sistemas de autenticação federados (MORGAN *et al.*, 2004), onde o sistema todo é mantido por uma única organização onde usuários afiliados podem reutilizar suas identidades (armazenadas de maneira centralizada) em

[5] Em (JOSANG; POPE, 2005), consideram-se como independentes as organizações que não fazem parte dos serviços onde a identidade do usuário é requisitada.

outros serviços pertencentes à mesma organização. É importante observar que esse sistema depende da existência de uma relação de confiança entre usuários e a organização que mantém o sistema de gerenciamento das identidades e, também, entre serviços e a organização.

Dentro do escopo deste trabalho, pretende-se abordar exclusivamente os SGIs centrados no usuário. Na Internet, o gerenciamento de identidade possibilitado por esse tipo de SGI é também referenciado como *Identity 2.0* (HARDT, 2005), (BATY; ATYCY, 2010), (VACCA, 2009), (MALIKI; SEIGNEUR, 2007). Para efeitos de simplificação do texto, a partir desse ponto, todas as menções ao termo "SGI" serão relacionados a esta categoria particular de sistema.

2.2.1 Gerenciamento de Atributos de Identidade

Uma funcionalidade comumente encontrada em sistemas de gerenciamento de identidades é a possibilidade de o usuário armazenar informações ou atributos pessoais (tais como nome do usuário, email, idade, etc) no serviço de identidades e associá-los à sua conta. Dessa maneira é possível mediante o consentimento do usuário, viabilizar o compartilhamento de informações com qualquer *site* ou serviço no momento do *login.*

2.3 Elementos básicos de um SGI

Um SGI possui essencialmente três elementos (VACCA, 2009):

- **Usuário**: Pessoa ou organização que possui uma ou mais identidades no sistema.
- ***Identity Provider – IdP*** (Provedor de Identidade): Responsável por gerenciar e emitir as asserções a respeito da identidade dos usuários aos provedores de serviço. Uma asserção pode ser vista como um conjunto de bytes (*token* ou credencial) autenticado pelo IdP (e.g., por meio de uma assinatura digital ou de um código de autenticação de mensagens) que pode ser utilizada para comprovar

que um usuário é o proprietário daquela identidade (ou de parte dela). Diversos IdPs podem coexistir em um mesmo ambiente.

- ***Relying Part – RP*** (Parte Dependente) ou *Service Provider* (Provedor de Serviço): Serviço ou *site* da Internet que recebe asserções emitidas por um *IdP* e autoriza um usuário de acordo com a sua política de acesso.

Um SGI centrado no usuário pode ser classificado de acordo com o seu modo de operação (BHARGAV-SPANTZEL *et al.*, 2007). São dois os tipos básicos de SGI: o primeiro tipo é definido como SGI com **foco na conexão** e o segundo como SGI com **foco na credencial**. Os **SGIs com foco na conexão** necessitam da conexão com o IdP no momento da comprovação da identidade para um RP. Nesse modo, uma asserção é gerada para cada transação. Já nos **SGI com foco na credencial**, não há a necessidade da conexão com o IdP no momento da comprovação da identidade; neste caso, as asserções geradas pelo IdP são armazenadas localmente no cliente e podem ser reutilizadas diversas vezes (BAUER; MEINTS; HANSEN, 2005).

2.4 Tecnologias de Gerenciamento de Identidade e o padrão OpenID

Existem diversas tecnologias para o provimento ou desenvolvimento de SGIs. Em (MALIKI; SEIGNEUR, 2007) e (VACCA, 2009) encontram-se as principais tecnologias relacionadas. Dentre elas, destacam-se: XRI/XRD (LABALME; LINDELSEE; WACHOB, 2005), ID/WSF Liberty Alliance (ALLIANCE, 2003), Shibboleth (MORGAN *et al.*, 2004), Microsoft CardSpace (MALINEN, 2006), OpenID (RECORDON; FITZPATRICK, 2007), Facebook Connect (STONE; ALTO, 2008), SXIP (MERRELS, 2006) e Higgins (RUDDY *et al.*, 2006).

Com exceção do OpenID (RECORDON; REED, 2006) e Facebook Connect (STONE; ALTO, 2008), nenhuma das tecnologias citadas tem sido amplamente adotada no ambiente Web da Internet (SENGUPT; RAJGARHIA, 2010). Dentre os motivos apontados para esta baixa adoção, destacam-se: complexidade da tecnologia, padrões fechados, problemas de usabilidade e centralização da identidade por serviços não necessariamente confiáveis por parte dos usuários e/ou provedores de serviços (DHAMIJA; DUSSEAULT, 2008),

(VACCA, 2009), (CAMERON, 2005). No caso do Facebook Connect, embora esta seja uma tecnologia de padrão fechado, sua ampla adoção na Internet como provedor de identidade ocorreu devido ao imenso número de usuários registrados em sua rede social[6] (LEBLANC, 2011), (STONE; ALTO, 2008).

Em especial, o OpenID surgiu como uma solução leve, simples e de padrão aberto para o gerenciamento de identidade na Internet (RECORDON; FITZPATRICK, 2007). O protocolo do OpenID permite que um usuário efetue *login* em diversos *sites* na Internet usando apenas um navegador Web padrão. Em dezembro de 2009 foram estimadas mais de um bilhão de contas OpenID habilitadas na Internet e cerca de nove milhões de *sites* com suporte a essa tecnologia (KISSEL, 2009). Dentre os principais SGIs na Internet usando OpenID pode-se citar: Google[7], Yahoo[8], MySpace[9], Aol[10], Verisign[11] e MyOpenID[12].

A ampla adoção do OpenID, aliada à participação menos significativa das demais tecnologias de padrão aberto na Internet, provê ao OpenID uma posição de destaque dentre as tecnologias de SGI na Web. De fato, de acordo com a análise de (LEBLANC, 2011), o OpenID é a opção mais adequada para a autenticação e login de usuários na Web atualmente devido à sua simplicidade e interoperabilidade com os protocolos atuais. Por esta razão, o OpenID será utilizado como base para o desenvolvimento da solução proposta nesse trabalho. Isso possibilitará a interoperabilidade com os sistemas atuais OpenID já implementados na Internet.

2.4.1 OpenID

Essa seção descreve, de modo resumido, o protocolo OpenID de acordo com a sua especificação 2.0 do protocolo de autenticação (RECORDON; FITZPATRICK, 2007). Por motivos de simplificação do texto, alguns detalhes da arquitetura e modos alternativos

[6] http://www.facebook.com
[7] http://www.google.com
[8] http://www.yahoo.com
[9] http://myspace.com/
[10] http://aol.com/
[11] http://pip.verisignlabs.com/
[12] http://myopenid.com

de funcionamentos foram omitidos ou simplificados. Detalhes do protocolo podem ser obtidos em (RECORDON; FITZPATRICK, 2007) e (RECORDON; REED, 2006).

Os elementos básicos utilizados da arquitetura do OpenID são:

- **Agente Usuário**: O navegador do usuário final, o qual é compatível com o protocolo HTTP (FIELDING *et al.*, 1999).
- **Identificador OpenID**: Na sua forma mais básica, o identificador do usuário é uma URL HTTP ou HTTPS, e é usado para identificar univocamente um usuário na Internet. Por exemplo, a URL http://myopenidprovider.com/joao123 pode ser um identificador OpenID para o usuário "*joao123*".
- **OpenID Provider (OP)**: O provedor de identidades (IdP) que emite as asserções comprovando que um usuário possui um determinado identificador.
- **Relying Party (RP)**: O serviço ou *site* na Internet que deseja obter a prova (asserção) de que o usuário controla um identificador.

No OpenID não há qualquer entidade central responsável por aprovar ou registrar OPs. Diversos IdPs podem existir na rede, gerenciados por diferentes entidades ou pessoas, e os usuários podem livremente escolher qual IdP desejam usar. Com isso, o OpenID é considerado uma solução **descentralizada**.

Para fins de ilustração do uso do protocolo, a seguir é apresentado um exemplo de fluxo de eventos e páginas no navegador do ponto de vista do usuário, considerando que o mesmo possui uma conta OpenID no servidor (OP) localizado em http://myopenidprovider.com:

1. O usuário acessa um *site* na Web com suporte ao OpenID, por exemplo www.wishlistr.com;
2. Ao clicar em "login", o usuário tem a opção de utilizar uma conta do próprio *wishlistr* ou uma conta OpenID;
3. O usuário seleciona a opção de login pelo OpenID e informa ao servidor onde possui a conta OpenID (http://myopenidprovider.com);

4. O navegador é redirecionado para a página de login do servidor OP em http://myopenidprovider.com;
5. O usuário autentica-se no servidor OP (métodos de autenticação não fazem parte do escopo do OpenID);
6. O servidor OP confirma, por meio da interface Web, se o usuário realmente deseja se autenticar em www.wishlistr.com com o identificador http://myopenidprovider.com/joao123;
7. Após a confirmação do usuário, o navegador é redirecionado novamente para o site www.wishlistr.com e o usuário é logado com o identificador "http://myopenidprovider.com/joao123", o qual pode ser considerado como o *username* do usuário no *site*.

De modo geral, o protocolo OpenID fornece um método para **provar que um usuário controla um identificador OpenID**. Essa comprovação é feita por meio de asserções geradas pelo OP ao RP. A autenticação de cada asserção é feita por meio de um código de autenticação de mensagens (*MAC – Message Authentication Code*) criado a partir de um segredo compartilhado estabelecido entre o OP e o RP em sua primeira interação. Do ponto de vista do RP, além da verificação do MAC, o RP também acessa a URL definida como identificador e procura por *tags* específicas do OpenID no conteúdo acessado. Essas *tags* indicam, além de outros parâmetros de segurança, quem é o OP responsável pelas asserções relativas ao identificador.

O fluxo básico de eventos entre os elementos do sistema durante o processo de autenticação do OpenID pode ser visto na Figura 6 e é explicado a seguir:

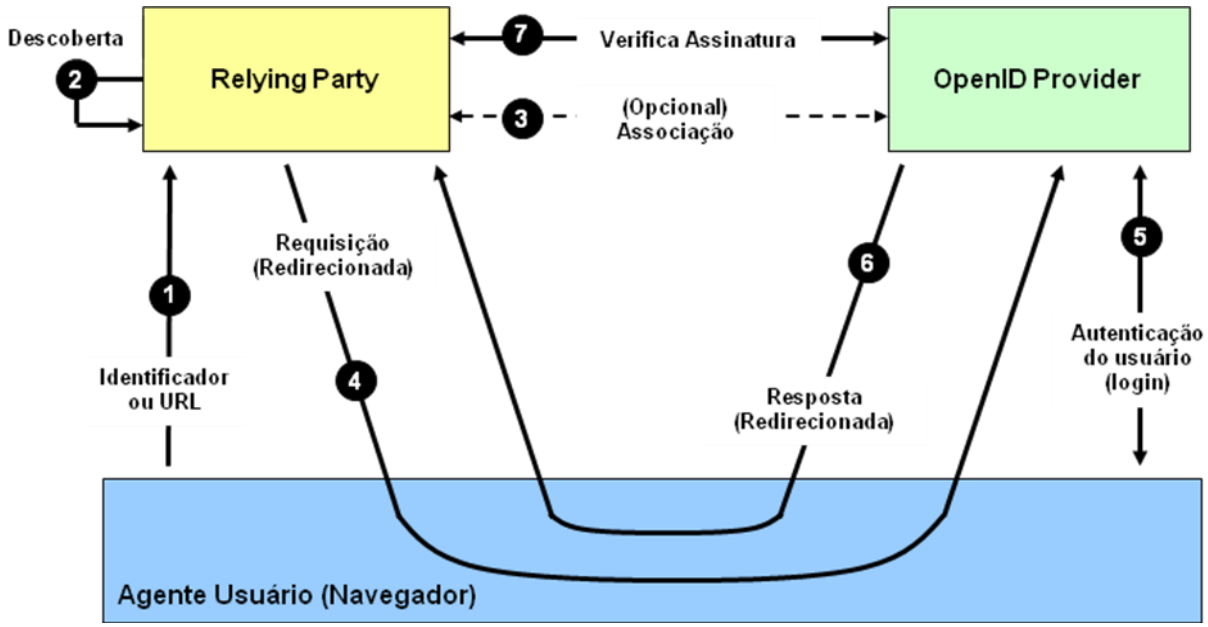

Figura 6 - Fluxo básico de eventos no OpenID

1. O usuário fornece ao RP o "identificador OpenID" ou a "URL do servidor OP" que abriga o seu identificador.

2. O RP inicia a descoberta da URL do OP, para onde o pedido de autenticação deve ser enviado. Essa descoberta é baseada na informação fornecida pelo usuário no passo 1.

3. Opcionalmente, o RP pode estabelecer um segredo compartilhado com o OP e armazená-lo localmente. Esse segredo é utilizado para a verificação das asserções geradas pelo OP na transação atual e, possivelmente, nas transações futuras.

4. O RP envia a requisição de autenticação do usuário para o OP. Essa requisição é feita via redirecionamento no navegador do cliente (mensagem HTTP) para a URL do OP. Os parâmetros da requisição de autenticação são inseridos na URL de redirecionamento e recuperados pelo OP.

5. O servidor OP autentica o cliente com base em algum tipo de credencial pré-estabelecida com o cliente (por exemplo, usuário e senha). O método de autenticação não faz parte do escopo da especificação do OpenID. É importante mencionar que esse passo pode ser suprimido caso o navegador do usuário já

esteja com a sua sessão autenticada. Devido a essa característica, o OpenID é visto também como um sistema de *Single Sign On* (RECORDON; REED, 2006).

6. Uma vez completado o procedimento de autenticação, o OP prepara a asserção do identificador do usuário e redireciona o navegador de volta para o RP. O RP, por sua vez, recupera a asserção da URL.

7. Finalmente, o RP verifica o código de autenticação de mensagem da asserção por meio da chave estabelecida no passo 3 ou por meio de um pedido de verificação enviado diretamente ao OP.

2.4.1.1 Relações de confiança entre usuário e OP no OpenID

No OpenID, o servidor OP precisa ser considerado confiável pelo usuário, uma vez que esse será o responsável por gerenciar os seus "identificadores OpenID". Um OP malicioso poderia utilizar esses identificadores em outros serviços, fazendo-se passar pelo usuário. Esse requisito é típico em sistemas de gerenciamento de identidades similares, uma vez que essa entidade é responsável pela emissão e gerenciamento das identidades dos usuários.

2.4.1.2 Extensões do OpenID

A especificação 2.0 do OpenID prevê extensões ao protocolo. Uma extensão tem a finalidade de permitir que informações extras sejam transmitidas juntamente com as mensagens regulares do protocolo. Como exemplo, pode-se citar a extensão *AX (Attribute Exchange,* ou troca de atributos*)* (HARDT; BUFU; HOYT, 2007), que permite que informações de identidade, tais como nome, email e data de nascimento, sejam trocadas entre OPs e RPs.

2.5 Resumo do Capítulo

Este capítulo apresentou os conceitos de identidade e SGI (Sistema de Gerenciamento de Identidades), de acordo com a literatura especializada na área. Foi visto também que

SGIs têm sido utilizados como uma solução para evitar o gerenciamento de contas múltiplas, principalmente na Internet. Neste contexto, o OpenID surgiu como uma solução leve e aberta, possibilitando a implementação de provedores de identidade na Internet. Com isso, o protocolo OpenID tem sido amplamente adotado na Web.

Embora tais sistemas de gerenciamento de identidade apresentem benefícios para o usuário, a sua implementação e adoção ampla na Internet pode causar um impacto negativo na privacidade dos usuários. Por meio das mensagens de autenticação que provam a existência de uma conta válida, um usuário pode ser associado à sua identidade real ou mesmo monitorado e rastreado na Internet por provedores de serviços e, principalmente, pelo sistema de gerenciamento de identidade. Com o intuito de melhor explorar este problema, o próximo capítulo introduz os conceitos de privacidade e anonimato, que serão utilizados ao longo deste trabalho, além de introduzir alguns mecanismos básicos utilizados no provimento destes serviços aos usuários.

3 PRIVACIDADE, ANONIMATO E CREDENCIAIS DIGITAIS

"I really believe that we don't have to make a trade-off between security and privacy. I think technology gives us the ability to have both."
John Poindexter – ex. conselheiro de segurança nacional norte-americano

Este capítulo apresenta os conceitos de **privacidade** e **anonimato** usados na literatura e adotados neste trabalho. É apresentado também o conceito de **credenciais digitais** como um mecanismo para possibilitar a privacidade do usuário. Além disso, são descritos, na forma de blocos construtivos, as técnicas criptográficas de assinaturas cegas (*blind signatures*) ou parcialmente cegas (*partially blind signatures*), usadas na criação das credenciais digitais.

3.1 Privacidade

De acordo com (WESTIN, 1967), privacidade (da informação) pode ser definida com o direito de um indivíduo, grupo ou instituição de determinar por si próprio quando, como, para quem e em que nível as informações sobre si são comunicadas a outros. Essa definição é amplamente usada na literatura para a discussão sobre a privacidade do usuário (BRANDS, S. A., 2000).

Aplicando-se esse conceito ao escopo deste trabalho, um SGI com suporte à privacidade deve permitir que um usuário decida quais informações (atributos de identidade) devem ser comunicadas a um serviço em particular durante uma tentativa de acesso. Em outras palavras, não deve ser possível para um serviço obter qualquer informação sobre o usuário além daquelas consentidas por este último.

3.2 Anonimato

De acordo com a taxonomia apresentada em (PFITZMANN; HANSEN, 2005), o **anonimato** de um indivíduo pode ser definido como o estado de ser **não-identificável**

dentro de um grupo de indivíduos (conhecido como *anonimity set*). Ainda segundo esta taxonomia, define-se a **não-relacionabilidade** (ou não-associabilidade) de dois ou mais itens de interesse da seguinte forma: dentro de um sistema contendo diversos itens, o conhecimento do relacionamento entre esses itens de interesse, que um atacante obtenha após observá-los, não deve ser nem maior nem menor do que conhecimento que ele tinha antes de observá-los.

É importante observar que a possibilidade de identificar uma pessoa em um grupo depende, entre outros fatores, do número de membros pertencentes àquele grupo. Em um exemplo extremo, onde o grupo é composto por apenas um indivíduo, a identificação do mesmo é imediata. Além disso, o comportamento do um indivíduo pode também comprometer o seu anonimato caso seja muito distinto dos demais (PFITZMANN; HANSEN, 2009). Tais fatores fazem com que não seja possível garantir o anonimato absoluto para um usuário em um sistema. É possível, porém, prover mecanismos de **melhoria** da **privacidade** do usuário e, dessa forma, **aumentar as chances** de existência de uma situação de **anonimato** (HEUPEL, 2010).

3.3 Credenciais Digitais para Suporte à Privacidade

De acordo com (HEUPEL, 2010) e (AULETE; VALENTE, 2006), uma credencial pode ser vista como um documento que pode ser utilizado por um indivíduo para convencer outro indivíduo sobre um determinado fato (por exemplo, um determinado atributo atestado na credencial).

Como ilustração, pode-se fazer uma analogia de uma credencial com um crachá de visitante dentro de uma empresa. Considerando um modelo de crachá para visitante onde é informado apenas o dia da visita (não sendo apresentada qualquer outra informação que identifique o visitante, tal como seu nome ou origem), seria possível para um observador inferir apenas que o usuário do crachá é um visitante dentro da empresa e que tem a permissão de visitação na data indicada no crachá. Em outras palavras, o crachá comprova dois atributos de seu usuário: o estado de “visitante” na empresa e a data em que a visita é válida.

Da mesma forma, uma credencial digital permite que um usuário comprove a veracidade de um ou mais atributos, de maneira digital, para uma entidade verificadora, de modo que **apenas** as informações necessárias são reveladas.

De acordo com a definição de (BRANDS, S., 2002), uma credencial digital, diferentemente de um certificado digital de chave pública, **não adiciona** "*identification handles*" (em tradução livre: marcas de identificação) que possam identificar um indivíduo dentro de um grupo. Isso implica que uma credencial digital pode, em sua essência, ser utilizada por um usuário para comprovar, de maneira anônima, que é detentor de certos atributos. Com isso, uma credencial deve prover mecanismos que dificultem o seu rastreamento (onde é utilizada) e também a sua associação com um usuário (que é o dono da credencial). Neste trabalho, o termo **credencial** será utilizado **unicamente** para referenciar tais credenciais.

Como outro exemplo, considere um aluno de uma universidade que deseja comprovar, por meio da Internet, seu estado de estudante para um cinema a fim de pagar meia-entrada no ingresso. O aluno deve ser capaz de obter uma credencial da universidade que comprove essa condição sem, porém, revelar qualquer outro atributo da sua identidade (o seu nome, por exemplo). Nesse cenário, a universidade não deve ser capaz de rastrear o aluno no uso da credencial. Essa propriedade torna-se possível se a credencial apresentada pelo usuário não possuir qualquer informação que o identifique dentre um conjunto de outras credenciais geradas pela universidade a outros alunos.

Diversas técnicas criptográficas podem ser empregadas para a geração de credenciais digitais (BRANDS, S., 2002). Este trabalho utilizará as técnicas de assinatura cega e assinatura parcialmente cega (CHAUM, 1983), (POINTCHEVAL; STERN, 1996), (CAMENISCH, J.; LYSYANSKAYA, 2003), (CAMENISCH, J.; GROSS, 2008), (CAMENISCH, J.; LYSYANSKAYA, 2001), apresentadas nas seções seguintes.

3.3.1 Assinaturas Cegas

O mecanismo tradicional de assinatura digital baseado em chaves públicas (DIFFIE; HELLMAN, 1976) consiste em uma função de assinatura privada ***S***, conhecida apenas

pelo assinante, e por um verificador público ***V***. Com isso, uma mensagem ***m*** qualquer assinada pelo assinante gera uma assinatura ***S(m)*** que pode ser verificada como correta pelo verificador caso ***V(S(m))*** seja verdadeira. É impraticável, nesse mecanismo, gerar a assinatura ***S(m)*** sem o conhecimento da função ***S***.

O mecanismo de assinatura cega (CHAUM, 1983) altera o tradicional mecanismo de assinatura digital com a introdução de uma função de ofuscamento **B** e a sua inversa B^{-1}, de modo que $B^{-1}(S(B(m))) = S(m)$. A função ***B*** é, ainda, tal que nenhuma informação de ***m*** pode ser obtida a partir de ***B(m).*** Considere as funções ***B*** e B^{-1} como conhecidas somente pelo emissor da mensagem ***m*** a ser assinada. Com isso, é possível para o emissor de ***m*** obter a assinatura ***S(m)*** sem revelar nenhuma informação sobre a mensagem ao assinante. Para isso, o emissor ofusca (*blinds*) a mensagem ***m***, gerando ***B(m)***, e envia para o assinante. O assinante, então, assina a mensagem ***B(m)*** e retorna para o emissor ***S(B(m))***. Finalmente, o emissor desofusca (*unblinds*) ***S(B(m))*** utilizando a função B^{-1}, obtendo assim a assinatura ***S(m)***.

Como uma simples analogia para o mecanismo de assinatura cega, pode-se imaginar um indivíduo (emissor) requisitando a outro indivíduo (assinante) para assinar, de olhos vendados, um documento cujo conteúdo é desconhecido para o assinante. Esse mecanismo possibilita a criação de algumas aplicações como, por exemplo, o de voto eletrônico (HORSTER; MICHELS; PETERSEN, 1995), onde o voto de um usuário permanece anônimo enquanto é, ao mesmo tempo, autenticado pela mesa eleitoral.

3.3.2 Assinaturas Parcialmente Cegas

Em algumas aplicações, o fato de o assinante desconhecer totalmente o conteúdo da mensagem a ser assinada é um fator limitador. Nesses casos, é necessário que o assinante possa, juntamente com o emissor, verificar e concordar com partes da mensagem (por exemplo, um conjunto de atributos) antes de assiná-la. Isso se aplica, por exemplo, na ilustração citada anteriormente em que o aluno de uma universidade deseja comprovar um atributo (“estudante matriculado”) a um cinema. Nessa ilustração,

é necessário que a universidade verifique e esteja de acordo com o atributo a ser assinado na credencial do aluno.

Dentro desse contexto, o conceito de Assinaturas Parcialmente Cegas foi introduzido de modo a possibilitar tais cenários (POINTCHEVAL; STERN, 1996), (CAMENISCH, J.; LYSYANSKAYA, 2003), (CAMENISCH, J.; GROSS, 2008), (CAMENISCH, J.; LYSYANSKAYA, 2001), (ZHANG; SAFAVI-NAINI; SUSILO, 2003a), (CHOW, S. *et al.*, 2005), (ZHANG; SAFAVI-NAINI; SUSILO, 2003a).

Esse mecanismo de assinatura parcialmente cega pode ser visto como uma variação da assinatura cega, onde a mensagem a ser assinada é composta por duas partes: ***m*** e ***info***. Considere ***m*** a parcela a ser ofuscada e desconhecida pelo assinante e ***info*** a parcela conhecida e de comum acordo entre o emissor e o assinante. Nessa variação, tem-se que $B^{-1}(S(B(m) \,||\, info)) = S(m \,||\, info)$.

3.4 Resumo do Capítulo

Este capítulo apresentou os conceitos de **privacidade** e **anonimato** que serão adotados ao longo deste trabalho. Mais precisamente, privacidade caracteriza-se como o direito do usuário em decidir a respeito de quais informações sobre sua identidade serão compartilhadas no acesso a um serviço. O anonimato, por sua vez, pode ser visto como a condição de não ser identificado dentre um conjunto de outros usuários (conjunto de anonimato). Com isso, pode-se dizer que, quando existe garantia de **privacidade**, é possível a um usuário obter acesso **anônimo** a um serviço desde que as informações compartilhadas pelo mesmo não sejam suficientes para identificá-lo dentro de um conjunto de usuários desse serviço.

Foi visto também que **credenciais digitais** podem ser utilizadas para o provimento de privacidade para um usuário. Por meio de tais credenciais é possível a comprovação de certos atributos de um usuário sem revelar informações (índices, números, códigos, chaves, etc.) que possam ser utilizadas para determinar a identidade do usuário e, consequentemente, outros atributos pessoais. Em outras palavras, pode-se afirmar que uma credencial digital pode comprovar atributos de um usuário **sem que sejam**

adicionadas quaisquer outras informações que possibilitem o rastreamento e a associação com a identidade real do usuário. Como visto também nesse capítulo, técnicas de assinaturas parcialmente cegas podem ser utilizadas para a criação de credenciais digitais.

Embora o uso de credenciais digitais em SGIs seja um caminho intuitivo para o gerenciamento de identidades com privacidade, a construção de tal solução no ambiente Web traz consigo desafios relacionados às características e requisitos de tal ambiente. Dessa maneira, o capítulo seguinte será destinado à discussão do problema e dos desafios envolvendo o provimento de identidades com suporte à privacidade na Web.

4 DESAFIOS DO AMBIENTE E ESPECIFICAÇÃO DE REQUISITOS

"The government cannot create that identity infrastructure[...] If it tried to, it wouldn't be trusted."

Jim Dempsey – Sobre a hipótese de criação de um provedor de identidades na Internet provido pelo governo norte-americano.

Este capítulo discute o ambiente Web na Internet e seus desafios para a construção de uma solução de SGI com suporte à privacidade. Serão também apresentados os requisitos de ambiente, privacidade e segurança desejados para tal solução.

Vale salientar mais uma vez que as questões de privacidade relativas às camadas de enlace, rede ou transporte no sistema não fazem parte do escopo desse trabalho e, por isso, não foram consideradas na discussão do problema ou no levantamento dos requisitos. Para essas camadas, outros mecanismos de anonimização por meio de redes *overlay*, como o sistema *Tor* (DINGLEDINE; MATHEWSON; SYVERSON, P., 2004), *Anonymizer* (BOYAN, 1997) ou (REITER; RUBIN, 1998) podem ser utilizados concomitantemente com a solução proposta.

4.1 Desafios do ambiente Web

Essa seção discute as principais características que devem ser respeitadas no desenvolvimento de um SGI na Web. Essa discussão é útil para a compreensão dos desafios do ambiente e posterior levantamento dos requisitos do sistema. Para melhor organização do texto, a discussão foi organizada em sub-tópicos, apresentados a seguir.

4.1.1 Confiança nos provedores de identidade por parte dos usuários

Uma vez que usuários na Internet não fazem parte de um mesmo grupo, organização, país ou cultura, é impossível exigir que um provedor de identidade, IdP, gerenciado por uma única organização (ou mesmo um grupo limitado de organizações) seja

considerado confiável por todos. Essa observação pode ser considerada válida mesmo em grupos específicos. Por exemplo, mesmo entre alunos de uma mesma universidade, não seria razoável inferir que todos concordariam em ter a universidade como gerenciador de identidades pessoais na Internet, uma vez que a mesma pode não ser considerada adequada por todos para essa finalidade. Logo, a arquitetura deve permitir a existência de IdPs de diferentes organizações, sendo possível para cada usuário escolher em qual IdP deseja confiar.

4.1.2 Verificação de uma identidade por parte dos *sites* ou serviços

Do ponto de vista de provedores de serviço, é fundamental que quaisquer tipos de credenciais para o acesso a um serviço sejam emitidos por entidades conhecidas e confiáveis de acordo com a política de acesso desse serviço. Em outras palavras, deve ser possível para um provedor de serviço aceitar credenciais baseando-se em uma lista de emissores reconhecidos por ele ou, alternativamente, definir as condições para o aceite de uma credencial.

Para efeitos de ilustração, considere um usuário que possui registro em três diferentes organizações que podem comprovar a sua idade: uma universidade, um banco e um clube. Considere também um *site* de jogos *on-line* que requer que o usuário seja maior de treze anos para o registro no serviço. Nesse cenário, o *site* poderia decidir aceitar credenciais assinadas pela universidade ou pelo banco para a comprovação de idade, porém não as credenciais assinadas pelo clube (por não possuir uma chave de assinatura atestada por uma CA confiável, por exemplo).

4.1.3 Manutenção do pseudônimo

Diferentemente de um sistema de pagamentos anônimo ou de voto eletrônico, onde não há a necessidade de se manter um histórico das atividades realizadas pelo usuário (por meio de um pseudônimo, por exemplo), o mesmo não acontece para um cenário como a Web. Nesse cenário, mesmo para acessos anônimos a um serviço ou *site*, é desejável que

o usuário tenha a opção de reutilizar, ou não, um mesmo pseudônimo. Isso permite, por exemplo, que o pseudônimo possa ser reconhecido pelo serviço (RP – *Relying Part*) como um usuário que **retorna** ao sistema e, com isso, usufruir de informações úteis salvas sobre suas sessões anteriores, ou mesmo da reputação construída com um determinado pseudônimo.

4.1.4 Uso da Web por meio de dispositivos distintos

Tipicamente os acessos de um determinado usuário aos serviços e *sites* na Web podem ocorrer a partir de equipamentos distintos ou mesmo compartilhados com outras pessoas. Por exemplo, o acesso à Web pode ser feito por meio do *notebook* no escritório, do *desktop* da residência ou mesmo de um computador de uso público em uma biblioteca ou *lan house*. Com isso, é desejável que uma solução de SGI para a Web não imponha o armazenamento local de dados ou credenciais do usuário (no dispositivo) como a única forma de possibilitar acessos subsequentes ao sistema.

Ainda, é importante lembrar que muitos dispositivos que acessam a Web possuem recursos computacionais de processamento e memória limitados. Dessa maneira, é importante que qualquer solução para esse ambiente mantenha-se o mais leve possível.

4.1.5 Tecnologias e protocolos Web

O ambiente da Web pode ser visto como uma plataforma composta de um conjunto limitado de tecnologias distintas (MADDEN; FOX, 2007). Logo, qualquer sistema que se destine a tal ambiente deve garantir a interoperabilidade com essas tecnologias.

Atualmente, a grande maioria dos acessos dos usuários a serviço e *sites* na Internet é realizada por meio de navegadores Web. Portanto, para que exista a interoperabilidade com o sistema atual é importante que um SGI para a Web opere também dentro dos limites das tecnologias existentes nos navegadores no lado do cliente (por exemplo, Javascript, Applets e/ou *plug-ins* do navegador).

O mesmo pode ser observado com relação aos protocolos utilizados na Web, como o HTTP e o HTTPS. A utilização de protocolos distintos pode causar problemas com possíveis *firewalls* e *proxies* localizados na rede do usuário ou em algum ponto entre o usuário e o serviço acessado. Além disso, como mencionado anteriormente na seção 2.4.1, diversos SGIs na Internet têm surgido com base no protocolo aberto OpenID. Para que o sistema seja interoperável com o sistema atual e, também, aumente suas chances de adoção na Internet, é interessante ainda que o mesmo seja compatível com o protocolo OpenID.

4.2 Especificação de Requisitos

Essa seção apresenta os requisitos levantados para a solução, separados em três grupos distintos: ambiente, privacidade e segurança. A maior parte dos requisitos é resultante da discussão das características do ambiente descritas anteriormente, enquanto outros se baseiam nos requisitos para SGIs federados apresentados em (BHARGAV-SPANTZEL *et al.*, 2007).

4.2.1 Ambiente

Os principais requisitos levantados para o cenário Web são listados a seguir:

I. **Escolha do IdP independente de organização:** Usuários devem poder escolher qual o IdP a ser utilizado, **independente** do serviço a ser acessado ou de quem é capaz de emitir credenciais.

II. **Possibilidade de acessos subsequentes a partir de equipamentos diferentes**: Deve ser possível para usuários acessar serviços ou *sites* na Internet a partir de lugares e equipamentos distintos ou mesmo compartilhados com outras pessoas. Isso implica que a solução não deve exigir o armazenamento local de dados ou credenciais como a única forma de possibilitar acessos subsequentes ao sistema.

III. **Uso de navegadores Web e tecnologias relacionadas no cliente:** O sistema deve ser compatível com as tecnologias existentes nos navegadores Web.

IV. **Possibilidade de reutilização de um pseudônimo em diferentes sessões:** Deve ser possível para um usuário a reutilização de um mesmo pseudônimo em acessos subsequentes em um serviço.

V. **Compatibilidade com protocolos existentes:** O sistema deve ser compatível com os protocolos de SGI atualmente utilizados na Web.

VI. **Descentralização:** Para a Internet, é desejável que um sistema seja descentralizado. A descentralização vai também ao encontro do requisito I, de modo a permitir que o sistema opere com diversos IdPs distribuídos na Internet e mantidos por diferentes organizações. A descentralização também permite que o sistema torne-se mais facilmente escalável.

VII. **Uso de recursos computacionais mínimos no cliente:** A demanda por processamento no cliente deve ser a menor possível, de modo a tornar mais favorável o acesso ao sistema por meio de dispositivos com baixo poder de processamento.

4.2.2 Privacidade

Esta seção apresenta os requisitos para uma solução de SGI com suporte à privacidade e comprovação de atributos de identidade:

I. **Não-relacionabilidade do usuário:** O sistema por si próprio não deve fornecer informações que possibilitem o estabelecimento da relação entre o usuário (identidade real) e seu acesso a um serviço por meio de um pseudônimo.

II. **Não-relacionabilidade entre acessos:** O sistema por si próprio não deve fornecer informações que possibilitem o estabelecimento da relação entre acessos distintos de um mesmo usuário em serviços.

III. **Comprovação de atributos selecionados pelo usuário:** O sistema deve permitir que o usuário comprove atributos de sua identidade a um serviço. A comprovação desses atributos necessita ser aprovado pelo usuário no momento de sua comunicação.

IV. **Minimização de dados (*data minimization*):** Deve ser possível a um usuário o fornecimento de apenas os dados necessários e suficientes para o acesso a um determinado serviço ou *site* na Internet.

4.2.3 Segurança

Esta seção lista os requisitos de segurança **específicos** para um SGIs. Alguns requisitos de segurança não são citados, como, por exemplo, disponibilidade ou prevenção a ataques de negação de serviço, por serem entendidos como desejáveis em qualquer serviço na Internet e, muitas vezes, implementados por meio de tecnologias não relacionadas ao protocolo ou à arquitetura do serviço.

I. **Confidencialidade:** O sistema e protocolos devem proteger informações sensíveis do usuário e relacionados à transação em si contra acessos não-autorizados.

II. **Prevenção a ataques de repetição:** O protocolo deve prevenir ataques de repetição (*replay*) na utilização das credenciais do usuário.

III. **Não-transitividade:** Deve ser impossível para um serviço ou *site* na Internet requisitar credenciais de um usuário com a finalidade de ganhar acesso em outro serviço no lugar do usuário.

IV. **Prevenção à personificação de pseudônimos:** Deve ser possível prevenir a reutilização de um pseudônimo por qualquer outro usuário diferente do detentor original do pseudônimo.

4.3 Resumo do Capítulo

Este capítulo discutiu as principais características que devem ser respeitadas no desenvolvimento de um SGI na Web e apresentou uma especificação de requisitos para a solução.

A discussão desse capítulo é importante, pois, apesar de soluções de gerenciamento de identidade com suporte à privacidade terem sido apresentadas anteriormente, nenhuma solução se propôs a enfocar as necessidades e limitações existentes no ambiente típico da interação entre usuários e *sites* no ambiente Web da Internet. Assim, a compreensão deste capítulo ajudará a tornar mais explícitas as diferenças entre o presente trabalho e trabalhos relacionados, apresentados no próximo capítulo.

5 TRABALHOS RELACIONADOS

Este capítulo apresenta uma visão geral dos trabalhos relacionados encontrados na literatura. Essa visão é importante para o conhecimento e entendimento do estado da arte de soluções na área de gerenciamento de identidades.

De modo a simplificar o texto e sua compreensão, os trabalhos são divididos em três grupos, de acordo com as suas características básicas e objetivos. Ao final deste capítulo, é apresentada uma tabela explicitando quais requisitos de sistema, conforme apresentado na seção 4.2, não são atendidos por estas soluções.

5.1 OpenID, Facebook Connect, Liberty Alliance, Cardspace, Sxip e Higgins

Este primeiro grupo de trabalhos preocupa-se basicamente com o gerenciamento de identidades de usuários conforme descrito no capítulo 2, sobre SGIs. Tipicamente, essas soluções suportam, também, o compartilhamento de atributos de identidade do usuário. Dentre os trabalhos mais importantes nessa categoria, pode-se citar: OpenID (RECORDON; FITZPATRICK, 2007), Facebook Connect (STONE; ALTO, 2008), ID/WSF *Liberty Alliance* (ALLIANCE, 2003), Microsoft CardSpace (MALINEN, 2006), SXIP (MERRELS, 2006) e Higgins (RUDDY *et al.*, 2006).

O fluxo básico de operação empregado nestas soluções é bastante semelhante entre os trabalhos que fazem parte desse primeiro grupo. De forma geral, após a tentativa de acesso de um **cliente**, o fluxo segue pela requisição de autenticação por parte do **serviço**. De posse dessa requisição, o **cliente** contata o **provedor de identidades** (IdP) e requer uma asserção sobre sua identidade e seus atributos. Uma vez recebida a resposta do **provedor de identidade**, o **cliente** entrega a asserção ao **serviço**, o qual pode efetuar a validação e prover acesso ao **cliente**.

Em especial, OpenID e Facebook Connect destacam-se por sua ampla utilização na Web. Diferentemente do OpenID, o Facebook Connect é um protocolo proprietário e de uso

exclusivo para o site de relacionamentos e provedor de identidades Facebook[13] (LEBLANC, 2011), (STONE; ALTO, 2008).

De modo geral, os trabalhos deste grupo não atendem aos requisitos especificados no capítulo anterior, pois não possuem nenhum mecanismo que garanta a privacidade do usuário. O detalhamento dos requisitos cumpridos por cada solução é apresentado na Tabela 1.

5.2 PseudoID, Canard e SPARTA

Neste segundo grupo, encontram-se os trabalhos que se concentram no **provimento de pseudônimos** com suporte à privacidade para os usuários do sistema. Tais pseudônimos são comprovados por meio de asserções não-rastreáveis e de uso único. Tipicamente, há uma relação de confiança estrita entre o **usuário** e o **provedor de pseudônimos**. Dentre os trabalhos mais importantes nesse grupo destacam-se o *PseudoID* (DEY; WEIS, 2010), *Canard* (CANARD; MALVILLE; TRAORÉ, 2008) e SPARTA (BIANCHI *et al.*, 2008).

O fluxo básico de operação dos trabalhos neste grupo difere-se do fluxo do grupo anterior no momento da obtenção da asserção do provedor de pseudônimos. Nos trabalhos deste grupo, o pseudônimo informado na asserção não é conhecido pelo provedor de pseudônimos. Isso permite que o usuário possa utilizar seu pseudônimo de modo anônimo no acesso a outros serviços dentro de uma mesma federação.

Apesar de fornecerem mecanismos de proteção à privacidade do usuário, de modo geral esses trabalhos não satisfazem aos requisitos do ambiente, como especificado na seção 4.2, e não possuem mecanismos para o gerenciamento e comprovação de atributos de identidade do usuário. Os detalhes para cada solução podem ser vistos na Tabela 1.

[13] http://www.facebook.com

5.3 U-Prove e Idemix

Este terceiro grupo de trabalhos caracteriza-se pelo provimento de um alto grau de anonimato para usuários por meio de algoritmos de *Zero-Knowledge Proofs* (FEIGE; FIAT; SHAMIR, 1988). Os trabalhos mais relevantes nessa categoria são Idemix (BICHSEL; CAMENISCH, J., 2010) e U-Prove (BRANDS, S. A., 2000).

De maneira geral, esses trabalhos possuem uma arquitetura de três elementos: **cliente**, **emissor** e **verificador**. O **emissor** é responsável pela geração de credenciais para o **cliente**, enquanto o **verificador** é o responsável por confirmar a autenticidade das informações apresentadas pelo **cliente** por meio da credencial obtida. De maneira geral, esses trabalhos apresentam dois protocolos, os quais são executados em momentos distintos no sistema. Em um primeiro momento, o **usuário** requisita uma credencial ao **emissor** e armazena essa credencial localmente. Com isso, o **usuário** pode, posteriormente, apresentar essas credenciais para obter acesso a serviços no sistema.

Essas credenciais armazenam diversos atributos da identidade do **usuário** que podem ser revelados para o **serviço** de maneira seletiva por meio do uso dos algoritmos de *Zero-Knowledge Proofs*. De modo geral, as credenciais podem ser reutilizadas e, por isso, demandam mecanismos de revogação de credenciais. Isso implica também que essas soluções demandam o armazenamento seguro das credenciais no dispositivo.

Em particular, o Idemix permite a criação de credenciais anônimas que podem ser "de-anonimizadas" por uma entidade confiável, caso uma atividade ilegal seja executada pelo dono da credencial anônima. Comparando estas duas soluções, o U-Prove provê um subconjunto das características definidas e especificadas pelo Idemix. Em outras palavras, embora ambos os sistemas possuam um funcionamento semelhante, o Idemix é de maneira geral mais complexo que o U-Prove (BICHSEL; CAMENISCH, J., 2010).

O U-Prove possui ainda uma limitação em relação ao provimento de privacidade ao usuário. Embora uma credencial U-Prove não possa ser diretamente associada com o seu usuário, é possível a associação entre os diversos usos de uma mesma credencial em diferentes serviços (CORELLA, 2011), (BRANDS, S. A., 2000).

Esses trabalhos são adequados para sistemas de identidade federados, onde as entidades do sistema e seus usuários fazem parte de uma mesma organização. Dessa forma, entre os principais requisitos não atendidos por esse grupo, pode-se citar a escolha do IdP independente, de acordo com as preferências do usuário. Ainda, devido ao uso de credenciais reutilizáveis armazenadas localmente, pode-se citar o não cumprimento do requisito de acessos subsequentes a partir de equipamentos diferentes.

5.4 Comparativo no atendimento aos requisitos

A Tabela 1, a seguir, apresenta uma análise comparativa referente ao atendimento aos requisitos de ambiente e privacidade, especificados no capítulo 4, pelas soluções mencionadas neste capítulo. Os requisitos de segurança não foram comparados na Tabela 1, pois todas as soluções discutidas cumprem esses requisitos.

Tabela 1 – Atendimento aos requisitos em cada um dos trabalhos relacionados

		Grupo 1					Grupo 2			Grupo 3	
		OpenID	Facebook	L. Alliance	Cardspace	Sxip	PseudoID	Canard	SPARTA	U-Prove	Idemix
Ambiente	Escolha do IdP independente de organização	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	Acessos subsequentes em equipamentos diferentes	✓	✓	✗	✗	✓	✗	✓	✓	✗	✗
	Uso de navegadores Web e tecnologias relacionadas	✓	✓	✗	✗	✓	✓	✗	✗	✗	✗
	Compatibilidade com protocolos existentes	✓	✓	✗	✗	✓	✓	✗	✗	✗	✗
	Reutilização de pseudônimos em diferentes sessões	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓
	Descentralização da solução	✓	✗	✓	✓	✗	✓	✓	✓	✓	✓
Privacidade	Não-relacionabilidade do usuário	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓
	Não-relacionabilidade entre acessos	✗	✗	✗	✗	✗	✗	✓	✓	✗	✓
	Minimização de dados	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓
	Comprovação de atributos selecionados pelo usuário	!	✗	!	!	!	✗	✗	✗	✓	✓

Legenda

✓ Atende

✗ Não atende / Não se aplica

! Atende parcialmente

Ao observar a Tabela 1, é possível perceber que grande parte dos **requisitos de ambiente** não é atendida pelas soluções, principalmente no grupo 3. Isto se deve ao fato de que diversas dessas soluções não foram construídas com o ambiente Web como foco principal. De fato, diversos enfoques distintos podem ser considerados para cada uma das soluções. Por exemplo, o U-Prove e Idemix foram criados com o enfoque nos algoritmos e mecanismos criptográficos necessários para a criação de credenciais com suporte à privacidade, sem um ambiente específico como objetivo. Já no Liberty Alliance, o enfoque foi dado para a criação de um sistema flexível em relação ao conteúdo de suas asserções. O Cardspace (criado pela Microsoft), por sua vez, foi criado com o enfoque de prover um sistema de gerenciamento de identidades integrado ao sistema operacional Windows.

Ainda, ao observar a Tabela 1, é possível perceber que as soluções do grupo 1, de modo geral, não suportam os **requisitos de privacidade** definidos para a solução. O requisito "comprovação de atributos selecionados pelo usuário" foi considerado como parcialmente atendido por algumas soluções devido ao suporte discutível nesse requisito. Por um lado, a comprovação de atributos é suportada por essas soluções, de modo que apenas os atributos selecionados são vistos por um serviço; por outro lado, é sempre possível para o provedor de identidades conhecer quais os atributos foram revelados pelo usuário e em quais serviços.

Por último, como mencionado anteriormente, as soluções do grupo 2 não prevêem mecanismos de comprovação de atributos pelo usuário. Isso pode ser visto na Tabela 1, com o não atendimento dos requisitos de "comprovação de atributos selecionados pelo usuário e "minimização de dados" (no caso deste último, o requisito não se aplica).

5.5 Resumo do Capítulo

Este capítulo apresentou uma visão geral dos trabalhos relacionados, destacando-se sua relação com os requisitos de sistema especificados. De maneira geral, pode-se afirmar que esses requisitos de sistema não são atendidos em sua totalidade por nenhum dos trabalhos relacionados neste capítulo, de acordo com a seção 4.2.

6 A ARQUITETURA E O PROTOCOLO NIBBLEID

" I don't think when people sign up for something they love they should have to sign up for a complete loss of privacy. I understand a little loss of privacy coming with the job."
Sarah Chalke – Atriz Canadense

Este capítulo propõe o NibbleID, um sistema para o gerenciamento de identidades com suporte a privacidade e comprovação de atributos na para Web, cujo projeto foi concebido de acordo com os requisitos levantados na seção 4.2. O NibbleID permite que um usuário selecione parte dos atributos de sua identidade e comprove-os para um provedor de serviços ou *site* na Web por meio de credenciais.

O suporte à privacidade do usuário no NibbleID foi concebido neste trabalho de acordo com a definição de privacidade, como apresentado na seção 3.1. Isso significa que o NibbleID permite que seus usuários apresentem **apenas** as informações (atributos) **consentidas** por eles para a utilização de um serviço. Isso implica a ocultação ou dissimulação de quaisquer outras informações que induzam a associação da identidade real do usuário com a identidade parcial apresentada e possibilitem o seu rastreamento. Com isso, o grau de anonimato de um usuário passa a depender de fatores externos à arquitetura e ao protocolo do sistema, como a natureza dos atributos compartilhados pelo usuário e a taxa de utilização no sistema. Uma discussão sobre os graus de anonimato de um usuário utilizando o NibbleID será apresentada posteriormente no capítulo 8.

O protocolo do NibbleID utiliza o **OpenID** como substrato para a comunicação entre usuário, provedor de identidade e provedor de serviço. O uso desse protocolo permite que o NibbleID seja em parte interoperável com serviços que suportam **OpenID** já estabelecidos na Web (nesse caso, sem a possibilidade de verificação e uso das credenciais por parte dos serviços). Uma análise formal do protocolo NibbleID será apresentada posteriormente no capítulo 7.

Este capítulo inicia-se com as definições utilizadas no NibbleID e, em seguida, apresenta a arquitetura e o protocolo da solução.

6.1 Definições utilizadas no NibbleID

Esta seção define os conceitos de Identidade, Identidade Parcial e Credencial, da forma como utilizados no NibbleID.

6.1.1 Identidade

Da mesma forma como apresentado na seção 2.1, uma identidade no NibbleID é compreendida como a representação de um usuário em um domínio por meio de um conjunto de atributos. A Figura 7 ilustra exemplos de três identidades distintas de um mesmo usuário em diferentes domínios: universidade, banco e governo.

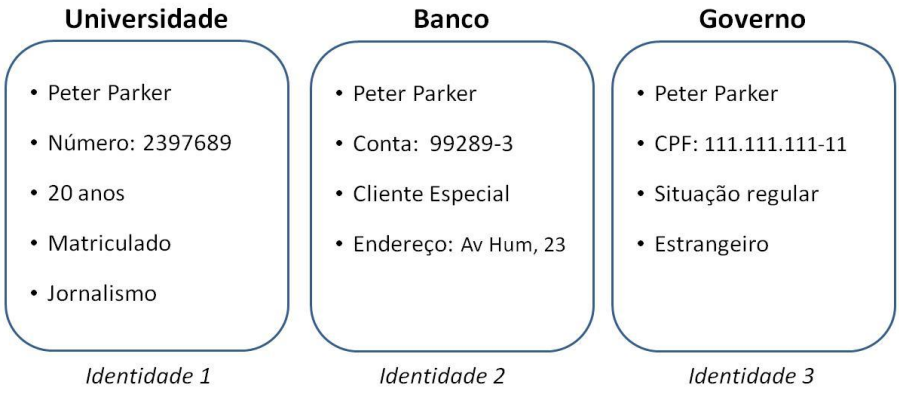

Figura 7 – Exemplo de três identidades de um mesmo usuário em domínios distintos

6.1.2 Identidade Parcial (ou Identidade Parcialmente Revelada)

De acordo com (PFITZMANN; HANSEN, 2009), uma identidade parcial é composta por um subconjunto dos atributos de uma identidade. De maneira similar, no NibbleID, uma identidade parcial é compreendida como um subconjunto de atributos provenientes de uma ou mais identidades de um mesmo usuário (possivelmente em domínios diferentes) que são reveladas para um serviço. Os atributos de uma identidade parcial são verificáveis por meio de credenciais (definidas a seguir). A Figura 8 ilustra exemplos

de três identidades parciais criadas a partir de subconjuntos de atributos provenientes de identidades de um usuário em domínios distintos.

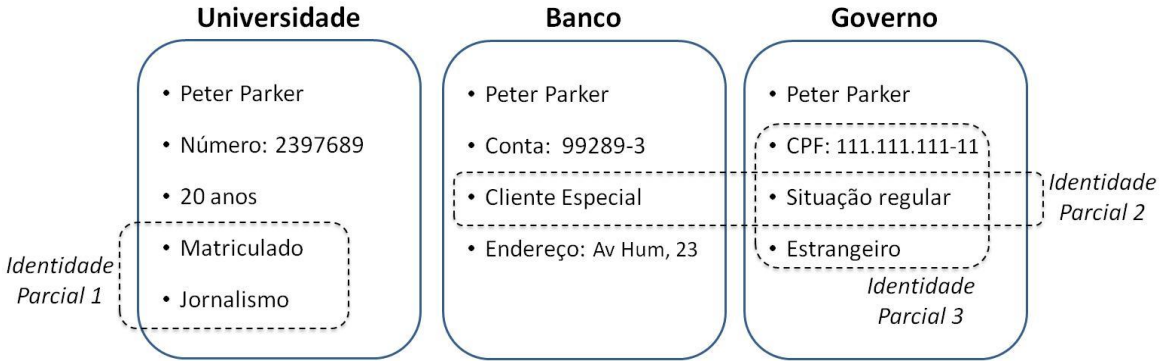

Figura 8 - Exemplo de identidades parciais criadas a partir de subconjuntos de atributos provenientes de diferentes identidades de um mesmo usuário

É importante perceber que o conceito de identidade parcial não leva em consideração a natureza dos atributos. Isso implica que uma identidade parcial pode, ou não, possuir atributos que a associem diretamente à identidade de um usuário em um domínio. Por exemplo, na Figura 8, a *identidade parcial 1* não é suficiente para identificar o aluno na universidade (considerando-se a existência de diversos outros alunos matriculados em jornalismo). Por outro lado, a *identidade parcial 3* identifica unicamente o indivíduo no sistema do governo (considerando-se CPF um atributo único do indivíduo).

Por este motivo, o conceito de **identidade parcial** pode ser também descrito como **identidade parcialmente revelada**. Porém, para efeitos de simplificação do texto e compatibilidade com a definição utilizada na literatura (PFITZMANN; HANSEN, 2009), o termo **identidade parcial** será adotado neste trabalho.

6.1.3 Credencial

De maneira similar à definição apresentada na seção 3.3, uma credencial no NibbleID é compreendida como um *token* que permite a **comprovação** atributos de um usuário em um determinado domínio e, ao mesmo tempo, **não adiciona** informações que possam

ser utilizadas para o rastreamento ou a associação desse usuário com a sua identidade real do domínio.

A utilização de credenciais, ao oposto de simples asserções assinadas, é vital para a preservação da privacidade do usuário, uma vez com tais credenciais é possível a exposição de apenas as informações **consentidas** pelo usuário a um serviço (de acordo com a definição de privacidade apresentada na seção 3.1 e a definição de credenciais digitais na seção 3.3).

6.1.4 Pseudônimo

De acordo com (PFITZMANN; HANSEN, 2005), um **pseudônimo** é um identificador de um sujeito que é diferente de seu nome real e pode ser utilizado para proteger sua identidade. De forma similar, um **pseudônimo** no NibbleID pode ser compreendido como um **identificador de usuário que não possui uma associação óbvia com uma identidade real**.

6.2 Arquitetura do NibbleID

Esta seção apresenta a arquitetura do NibbleID, explicitando as entidades do sistema e suas interações. São apresentadas também as estruturas de dados fundamentais, tais como pseudônimos, identidades parciais e credenciais.

6.2.1 Elementos da Arquitetura

Sistemas de gerenciamento de identidade possuem tipicamente três elementos em sua arquitetura: o **cliente**, o **serviço** e o **provedor de identidades** (VACCA, 2009). Nesses sistemas, de modo geral, é de responsabilidade do provedor de identidades o gerenciamento e a emissão de asserções ou credenciais a respeito dos **pseudônimos** e **atributos de identidade** do usuário.

Esse modelo de sistema de gerenciamento de identidades tipicamente não permite que o usuário escolha o provedor de identidades de sua preferência independente de federação. Nesse modelo, o provedor de identidades é, também, o emissor das asserções ou credenciais, de forma que o usuário é obrigado a ter sua identidade provida pela mesma organização que possui registrados os seus atributos de identidade.

Além disso, esse modelo com três entidades apresenta outra limitação na geração de identidades no ambiente Web: o provedor de identidades pode somente atestar os atributos de identidade que são conhecidos e verificáveis por ele. Por exemplo, considere o gerenciamento de identidade sendo provido por uma universidade. Neste exemplo, a identidade fornecida pela universidade é limitada a conter informações conhecidas por ela (e.g., nome do aluno, notas, matrícula ativa). Assim, outras informações que sejam pertinentes a outras instituições, tais como bancos ou governo (e.g., se usuário é possuidor de conta corrente, situação eleitoral regular, CPF válido) não podem ser certificadas, em princípio, e associadas à mesma identidade que foi provida pela universidade.

A fim de evitar estas limitações, a arquitetura do NibbleID separa as funções de provimento de identidades em duas funções: **provimento de pseudônimo** e **provimento de credenciais**. Para isto, a arquitetura define dois elementos distintos: o ***Identity Provider (IdP)***[14] para o provimento de pseudônimos e o ***Credential Provider (CP)*** para o provimento de credenciais. A arquitetura permite que estas duas entidades sejam partes de organizações distintas, de modo que um usuário pode escolher um provedor de identidades na Internet que seja independente das entidades habilitadas a certificar seus atributos (e.g., bancos, universidades e governo). Os nomes das entidades foram definidos em inglês para que alguns dos elementos tenham uma melhor relação com os nomes utilizados na literatura especializada. A seguir são descritos os elementos da arquitetura do NibbleID:

[14] O nome "*Pseudonym Provider*" (Provedor de Pseudônimo) foi também considerado para essa entidade, uma vez que a mesma não é detentora da identidade real ou completa do usuário (apenas o pseudônimo). Porém, o nome *Identity Provider* foi escolhido por ser mais comumente utilizado na literatura e ajudar na associação do papel exercido por essa entidade e o provedor de identidades da arquitetura do OpenID.

- ***User*** (usuário): Usuário que acessa o sistema por meio de um navegador padrão (cliente). Opcionalmente, um *plug-in* pode ser utilizado no navegador para a melhoria da segurança e usabilidade do sistema. Neste trabalho, ambos os termos *usuário* ou *cliente* são utilizados para descrever o elemento *User* no sistema.

- ***Credential Provider – CP*** (Provedor de Credenciais)**:** Entidade responsável pela emissão de credenciais para o usuário. Um usuário pode ter um ou mais *CPs* (e.g. universidade, governo e banco). O *CP* deve ser capaz de autenticar o usuário e conhecer atributos de sua identidades. Para a preservar da privacidade do usuário, dentro de certas condições (estudadas adiante no capítulo 8), não é possível para o CP rastrear onde (em quais serviços) as credenciais emitidas para um cliente foram utilizadas.

- ***Identity Provider – IdP*** (Provedor de Identidades): Entidade responsável pelo gerenciamento de pseudônimos utilizados pelo usuário para identificar cada uma das identidades parciais que o usuário possui junto aos *sites* ou serviços na Web. Em princípio, o *IdP* não deve ser capaz de associar o pseudônimo do usuário à(s) identidade(s) registrada(s) em CP(s), salvo o caso em que o usuário tenha deliberadamente fornecido informações que permitam essa associação. As funções do IdP são baseadas no *OP (OpenID Provider)* da arquitetura OpenID. Esse provedor deve poder ser livremente escolhido pelo usuário.

- ***Relying Party – RP*** (Provedor de Serviço): Provedor de serviço ou *site* na Internet que provê acesso a um usuário baseado em atributos de identidade[15]. Cada RP pode possuir uma lista de CPs aceitos ou estabelecer condições para a aceitação de uma credencial.

[15] Essa forma de prover acesso é também conhecida na literatura como “autorização sem autenticação” (KARAGODIN, 2005).

6.2.2 Tipos de Pseudônimos

Por meio do NibbleID, um usuário pode utilizar **pseudônimos** para a representação de identidades parciais frente a um RP. Um **pseudônimo** deve ser criado pelo usuário em um IdP e não deve ser revelado a qualquer de seus CPs. Dessa maneira, embora a identidade real de um usuário seja conhecida por CP, não são conhecidos os **pseudônimos** utilizados por esse usuário.

Um **pseudônimo** é um identificador OpenID (como descrito em 2.4.1) e pode, por exemplo, assumir a forma "*http://idp.larc.usp.br/id/**meu_pseudônimo***" (considerando-se um IdP localizado em "*http://idp.larc.usp.br*"). Seguindo a especificação do OpenID, a comprovação de que um usuário possui um determinado **pseudônimo** em um IdP é feita por meio de asserções geradas por este IdP.

O NibbleID prevê três tipos distintos de **pseudônimos** que podem ser escolhidos pelo usuário junto ao IdP no momento do uso do serviço. Os três tipos são apresentados a seguir:

a. **Pseudônimos globais**: Pseudônimos globais são criados pelo usuário no IdP, e podem ser utilizados em diferentes *sites* na Web de maneira que o **pseudônimo** é imutável independentemente do RP acessado. Esse tipo é útil quando o usuário deseja ser reconhecido por meio de seu pseudônimo em diferentes *sites* na Internet. Por exemplo, um usuário pode decidir utilizar essa classe para construir algum tipo de reputação sobre o seu **pseudônimo,** quando escrever mensagens em diferentes ***blogs***, fóruns e *sites* na Internet. A desvantagem desse tipo de **pseudônimo** é a possibilidade de rastreamento de seu uso nos diversos RPs. A Figura 9, a seguir, ilustra a utilização de um **pseudônimo global** (localizado em "*http://idp.larc.usp.br*") em RPs distintos.

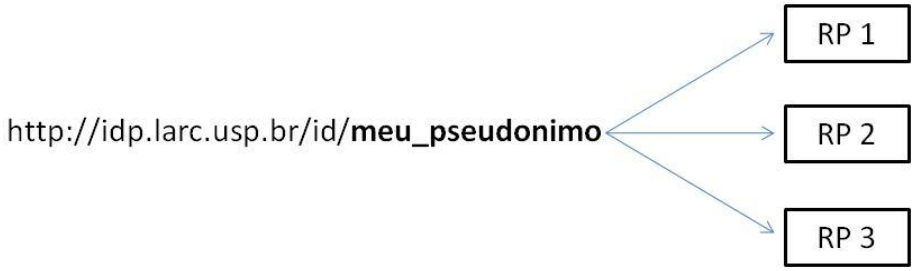

Figura 9 – Uso de um pseudônimo global em diversos RPs

b. **Pseudônimos específicos por RP**: Essa classe de pseudônimos pode ser utilizada quando o usuário deseja evitar a correlação entre acessos em diferentes *sites* na Internet. No caso de **pseudônimos globais**, essa correlação ocorre de maneira bastante simples, uma vez que o mesmo **pseudônimo** é utilizado para os diferentes RPs. **Pseudônimos específicos por RP** podem ser gerados pelo usuário ou pelo próprio IdP com base na URL do *site* e alguma informação fixa do usuário (por exemplo, o nome do usuário OpenID). Cada pseudônimo é utilizado em um RP específico. Como exemplo, considere na Figura 10 diversos pseudônimos de um **mesmo usuário** localizados em um IdP "*http://idp.larc.usp.br*". Neste exemplo, o pseudônimo "*http://idp.larc.usp.br/id/i3bf86y7ls5g*" é um pseudônimo específico para o *site* "RP 1". Esse pseudônimo pode ser reutilizado em todos os acessos ao *site* "RP 1", o que possibilitaria este *site* detectar que o usuário está retornando ao sistema.

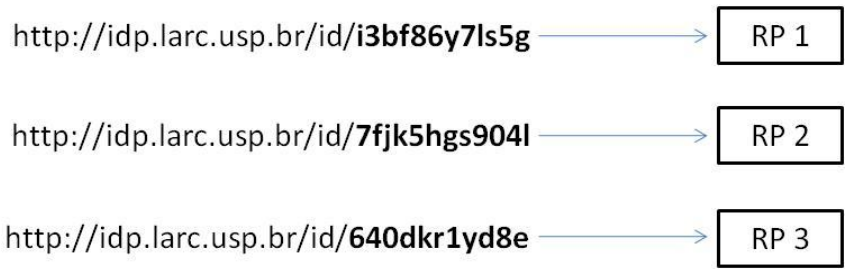

Figura 10 – Uso de pseudônimos específicos por site RP

c. **Pseudônimos descartáveis**: Esta classe de pseudônimos pode ser utilizada quando o usuário não deseja manter qualquer vínculo entre um acesso a um RP e os seus acessos subseqüentes (no mesmo RP ou em outros). **Pseudônimos descartáveis aleatórios** podem ser gerados pelo IdP (ou pelo cliente) sob demanda e descartados logo após o seu uso. A Figura 11 ilustra o uso de um **pseudônimo aleatório** gerado para o acesso único a um RP. É importante ressaltar que o IdP deve garantir que os pseudônimos não sejam repetidos em usos subseqüentes de um usuário ou mesmo por outros usuários.

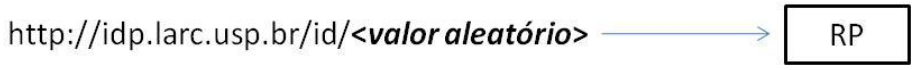

Figura 11- Uso de pseudônimo descartável para evitar a correlação entre acessos subsequentes

6.2.3 Comprovação de uma Identidade Parcial

Como definido anteriormente, uma **identidade parcial** é compreendida como um **subconjunto de atributos apresentados a um serviço** e é **identificada por um pseudônimo**. Os **atributos desta identidade parcial** são **comprovados por meio de credenciais** emitidas por um ou mais **CPs**, as quais são **associadas** (por meio das assinaturas parcialmente cegas) **ao pseudônimo da identidade parcial**. Para ilustrar, a Figura 12 mostra as relações entre **identidade parcial**, **pseudônimo** e **credenciais.** Na figura é possível verificar que uma identidade parcial é associada a um pseudônimo e pode ser associado a diversas credenciais. Esse conjunto de informações é apresentado a um serviço para a comprovação da identidade parcial de um usuário.

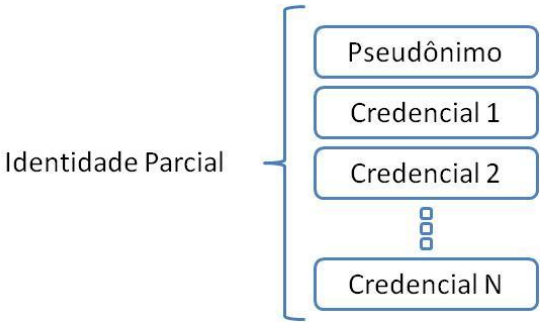

Figura 12 – Relação entre identidade parcial, pseudônimo e credenciais

No NibbleID**,** o número de atributos de uma identidade parcial pode crescer a partir da apresentação de novos atributos comprovados como parte de um mesmo pseudônimo. Com isso, **é possível a construção de uma identidade parcial composta por atributos provenientes de diferentes CPs**, como será visto adiante na seção 6.2.5.

6.2.4 Formato de uma Credencial

A Tabela 2, a seguir, apresenta os campos de uma credencial e a condição de cada campo durante a sua geração. A condição refere-se ao mecanismo de assinaturas parcialmente cegas, onde um campo pode ser ou não ofuscado[16] para a assinatura de CP.

[16] Um valor ofuscado não pode ser visto por CP, conforme o mecanismo de assinaturas parcialmente cegas descrito em 3.3.2.

Tabela 2 - Campos da credencial no NibbleID

Campo	Descrição	Condição na geração da credencial
Pseudônimo	Pseudônimo da identidade parcial	Ofuscado
Nonce	Número aleatório de uso único gerado por RP	Ofuscado
Atributo(s)	Atributo(s) do usuário certificado(s) por CP	Não ofuscado
Assinatura de CP[17]	Assinatura de CP sobre os campos anteriores	Ofuscado (a ser desofuscado pelo cliente)

O pseudônimo, por ser ofuscado, não é de conhecimento de CP, o que provê melhor privacidade para o usuário no uso da credencial. O mesmo ocorre com o *nonce*, gerado por RP para a garantia de uma credencial de uso único. Caso esses campos não fossem ofuscados, serviriam como informação óbvia para uma possível associação entre a credencial e a identidade do usuário em CP.

De maneira similar, a assinatura gerada por CP sobre os campos da credencial (com valores ofuscados) é posteriormente processada pelo cliente de modo a obter a assinatura sobre os parâmetros desofuscados. Isto impede que a assinatura de CP também seja utilizada para o rastreamento da credencial. Isso é possível devido à característica do mecanismo de assinaturas parcialmente cegas discutida na seção 3.3.2, dado por $\boldsymbol{B^{-1}(S(B(m) \,||\, info)) = S(m \,||\, info)}$. No NibbleID temos que:

- **B** e $\boldsymbol{B^{-1}}$: Funções de ofuscamento e desofuscamento conhecidas somente pelo usuário;
- **m:** *Pseudônimo* e *Nonce* concatenados (ofuscados pelo usuário antes da requisição de credencial. Não são observados por CP);
- **Info:** atributo(s) do usuário (conteúdo conhecido e de comum acordo entre usuário e CP);

[17] É suposto o uso de uma ICP (Infra-estrutura de Chaves Públicas) para a emissão do certificado de chave pública para CP de modo que essa entidade possa assinar as credenciais digitalmente.

- **S:** Assinatura de CP (gerado por CP sobre o conteúdo ofuscado e, posteriormente, desofuscado pelo usuário, resultando na assinatura de "pseudônimo||nonce||atributo(s)" ou "m||info").

6.2.5 Agregação de atributos a uma identidade parcial já existente

Uma vez que toda **identidade parcial** é associada a um **pseudônimo,** um usuário (dono do pseudônimo) pode agregar **atributos em uma identidade parcial** já existente por meio deste pseudônimo. O **processo de agregação** de atributos a uma identidade parcial existente ocorre por meio da apresentação de uma nova credencial (contendo os novos atributos) a um serviço. Essa nova credencial deve obrigatoriamente reutilizar o pseudônimo inicialmente associado à identidade parcial original. Esse **procedimento de agregação** de atributos pode ser efetuado para pseudônimos globais ou específicos por RP (apresentados na seção 6.2.2). Essa operação não é possível para pseudônimos descartáveis, uma vez que o pseudônimo é limitado a uma utilização somente.

É importante perceber que, para pseudônimos globais, os atributos de uma identidade parcial (associados a um pseudônimo) conhecidos em um RP podem ser diferentes dos atributos conhecidos por um outro RP. Isso depende dos atributos que foram informados pelo cliente durante os acessos a cada RP ou de uma possível troca de informações entre RPs (por exemplo, RPs de uma mesma organização).

6.2.6 Modos de requisição e geração de credenciais

O NibbleID permite dois modos distintos para a requisição e geração das credenciais. A escolha pode ser estabelecida de maneira estática pela implementação do protocolo ou, alternativamente, efetuada pelo cliente no momento da requisição da credencial (caso suportado pela implementação). Os modos são:

- **Atributos múltiplos por credencial**.
- **Atributos únicos por credencial**.

Um critério para a escolha do modo de requisição em geral se baseia na relação “custo computacional” versus “nível de privacidade” para o usuário, uma vez que a diferença básica entre esses modos está nesta relação. Estes dois modos são apresentados e explicados a seguir.

6.2.6.1 Atributos múltiplos por credencial

Neste modo, uma mesma credencial pode conter diversos atributos de um usuário. Para isso, os **atributos a serem revelados** a um RP são explicitamente informados pelo usuário ao CP no momento da requisição da credencial. Com esse conhecimento, CP agrupa todos os atributos como uma única informação e as insere em uma única credencial, conforme ilustrado na Figura 13.

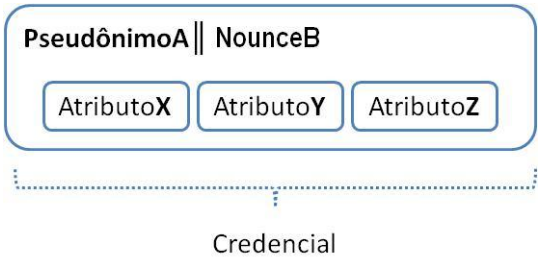

Figura 13 - Atributos múltiplos por credencial

A vantagem deste modo é a simplificação do processo (menor número de operações) uma vez que apenas uma credencial é gerada. Por outro lado, este modo traz um impacto negativo na privacidade do usuário, pois permite que o CP conheça os atributos que serão apresentados para o RP durante o uso da credencial. Isto ocorre porque, na apresentação de uma credencial, todos os seus atributos são revelados.

6.2.6.2 Atributos únicos por credencial

Neste modo, uma credencial contém apenas um atributo do usuário. Com isso, o CP deve gerar diversas credenciais, uma por atributo, a fim de permitir que o usuário selecione quais credenciais deseja apresentar posteriormente a um RP. Desta maneira, o CP não possui a informação sobre quais atributos o cliente apresentará para RP, o que resulta em maior privacidade para o usuário.

Por outro lado, este modo requer um número maior de operações no protocolo para a geração individual de cada credencial. A Figura 14 ilustra as credenciais geradas neste modo.

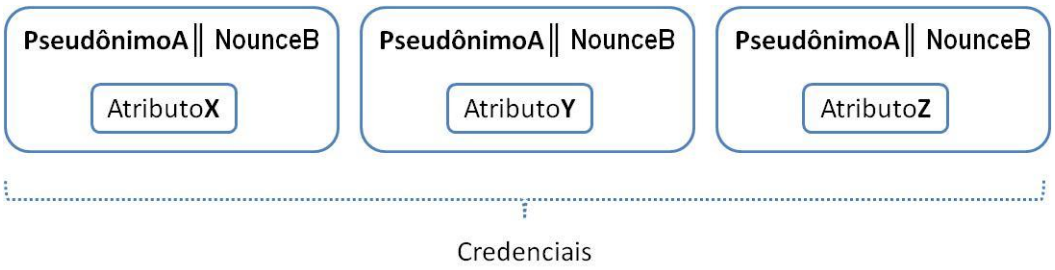

Figura 14 - Atributos únicos por credencial

6.3 Protocolo do NibbleID

Esta seção apresenta o protocolo do NibbleID. Para um melhor entendimento do protocolo, será apresentada primeiramente uma visão geral da seqüência de mensagens trocadas em **um fluxo completo de requisição e uso de uma credencial** (Figura 15). Em seguida, de maneira mais concisa, serão apresentadas as mensagens do protocolo (Tabela 4 e Tabela 6) para os dois modos de requisição e geração de credenciais previstos, explicitando-se as principais diferenças com relação ao fluxo básico de mensagens.

Para a simplificação dos fluxos, considerou-se o usuário já autenticado pelo IdP (i.e., considera-se a aplicação do *Single Sign On* do OpenID).

6.3.1 Visão geral

A Figura 15, a seguir, apresenta uma visão geral do protocolo por meio de um fluxo simplificado. Os valores indicados entre parêntesis (M1 a M14) foram incluídos para facilitar a associação das mensagens do fluxo simplificado com as mensagens da Tabela 4 e Tabela 6. O modo de requisição de credenciais com atributos múltiplos por credencial foi utilizado neste fluxo da visão geral devido a sua maior simplicidade.

Considera-se o uso de HTTPS entre o usuário e os servidores RP, IdP e CP. O uso de um canal seguro evita um possível sequestro da sessão do navegador e personificação do usuário (BERGHEL, 2002).

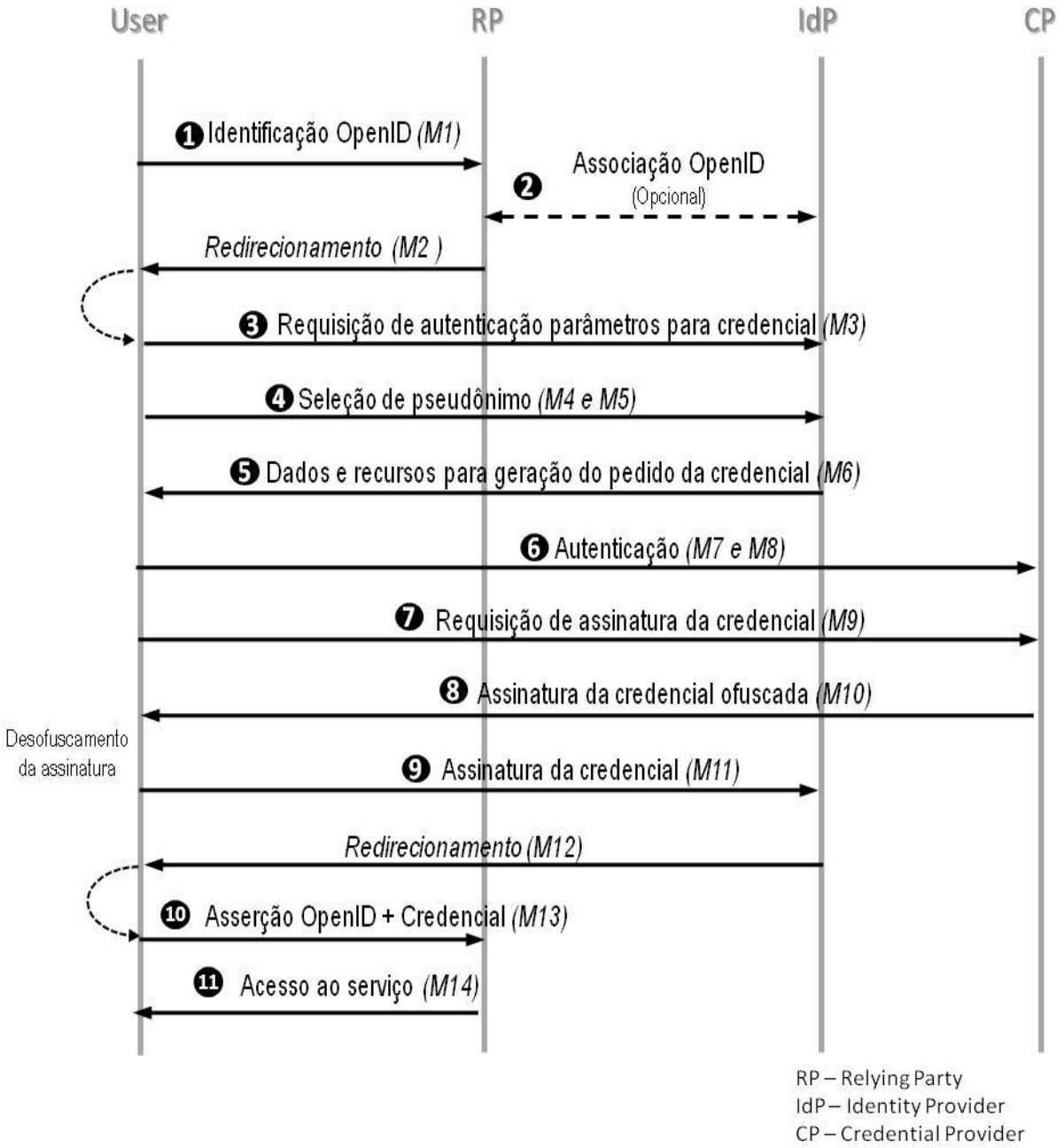

Figura 15 - Fluxo básico do protocolo proposto

O fluxo simplificado é explicado a seguir, de acordo com os números das mensagens na figura:

1. O usuário fornece o **identificador OpenID** para o **RP**. Esta mensagem é no padrão OpenID.

2. Opcionalmente, o RP pode estabelecer um segredo compartilhado entre IdP e armazená-lo localmente. Este passo é opcional e, também, padrão no OpenID.

3. O RP envia a **requisição de autenticação OpenID (**por meio de um redirecionamento HTTP). Além dos campos tradicionais do OpenID, contendo o endereço de IdP e RP, esta mensagem deve conter também os parâmetros para a geração de credenciais. Esses parâmetros devem ser enviados por meio de uma extensão do protocolo OpenID, assim como previsto por (RECORDON; FITZPATRICK, 2007) e mencionado na seção 2.4.1. De forma geral, os parâmetros de geração são:

    a. Atributos necessários para o acesso: Por exemplo, se o usuário é aluno matriculado em uma escola ou se sua idade é maior do que um mínimo determinado.

    b. Um valor aleatório único (*nonce*): O *nonce* deve ser utilizado na geração da credencial para garantir o seu uso único.

    c. Lista de CPs aceitos: Indicação de quais CPs são aceitos pelo RP para a geração da credencial.

4. Por meio da interface HTML do IdP, o usuário escolhe qual o pseudônimo deve ser informado ao RP (o usuário pode possuir diversos pseudônimos entre as três possíveis classes descritas na seção 6.2.2). Essa etapa pode ser utilizada também para a verificação visual e aprovação, por parte do usuário, dos atributos requeridos para a credencial.

5. O IdP envia os parâmetros necessários para a requisição da credencial anônima (pseudônimo, *nonce*, atributos da credencial e CP escolhido). Esta mensagem

possui também a função de prover ou disparar o *software*[18] no navegador para a realização do procedimento de assinatura cega.

As etapas 6, 7, 8 e 9 são realizadas no lado do usuário por meio do *software* provido ou disparado na etapa 5.

6. O usuário contata o CP para autenticação. Esta autenticação é baseada em algum tipo de relação de confiança preestabelecida entre o usuário e o CP (por exemplo, usuário e senha). O método de autenticação utilizado não faz parte do escopo do protocolo NibbleID e deve ser definido por CP (mais detalhes na seção 6.3.2).

7. Uma vez autenticado, o *software* no cliente gera as funções **B** e $\mathbf{B^{-1}}$ (baseados em valores aleatórios conhecidos somente pelo usuário). Em seguida, os seguintes parâmetros são enviados para o CP na requisição de assinatura da credencial:

    a. Pseudônimo e *nonce* ofuscados: ***B (pseudônimo||nonce)***

    b. Atributos requeridos para a credencial: ***info***

8. CP envia uma resposta contendo a assinatura de *B (pseudônimo||nonce)||info*, ou seja, ***S(B(pseudônimo||nonce)||info)***. Caso seja necessário, para efeitos de contabilização, CP pode incrementar um registro do número de credenciais geradas para aquele usuário.

9. O *software* no cliente desofusca a mensagem *S(B(pseudônimo||nonce)||info))* por meio da função $\mathbf{B^{-1}}$ (conhecida apenas pelo usuário), verifica os parâmetros recebidos pelo CP e envia a assinatura da credencial anônima para o IdP: ***S(pseudônimo||nonce||info)***.

10. O IdP prepara a asserção OpenID, adiciona os campos da credencial na mensagem de retorno e redireciona o navegador do usuário de volta ao RP. Os seguintes valores estão contidos na asserção OpenID e campos de extensão:

[18] O servidor pode prover código para ser executado no lado do cliente de diversas maneiras. Como exemplo, podem-se citar: *Java Applet, Javascript Flash ou ActiveX*. Alternativamente, o navegador pode já ter uma aplicação de assinatura cega pré-instalada (no caso de um *plugin* de navegador). Neste caso, esta aplicação pode ser disparada por uma mensagem vinda do servidor.

a. **Pseudônimo**: Identificador OpenID do usuário autenticado verificável pela asserção OpenID;

b. **Nonce**: Valor único gerado pelo RP para a credencial;

c. **Info**: Conjunto de atributos do usuário;

d. **Identificação do CP**: Um identificador para o CP gerador da assinatura ou o seu certificado de chave pública;

e. **Identificação do RP**: Endereço do RP que requisitou a credencial;

f. ***S(pseudônimo||nonce||info)***: Assinatura do conjunto que representa a credencial;

g. **MAC**: Código de autenticação da asserção gerada por IdP, de acordo com a especificação do OpenID.

11. Finalmente, o RP verifica a asserção OpenID e a assinatura da credencial gerada por CP e autoriza o acesso do usuário ao serviço. Essa autorização é baseada nas informações contidas na credencial e na política de acesso do RP.

6.3.2 Autenticação do usuário em IdP e CP

Da mesma forma que o OpenID não determina o método de autenticação a ser utilizado entre IdP e o usuário, o NibbleID também não determina os métodos de autenticação a serem utilizados entre o usuário e as entidades IdP e CP. Portanto, cabe ao IdP e ao CP a escolha de um método de autenticação apropriado às suas políticas de segurança. Um conjunto de recomendações e direções sobre o assunto pode ser encontrado em (BURR; DODSON; POLK, 2004).

6.3.3 Mensagens do protocolo para o modo de atributos múltiplos por credencial

Esta seção apresenta, de maneira concisa, as mensagens do protocolo para o modo de atributos múltiplos por credencial. A Tabela 3, a seguir, apresenta as notações utilizadas na descrição das mensagens.

Tabela 3 - Notações utilizadas para a descrição das mensagens no NibbleID

Notação	Significado
<X>	Resumo criptográfico da mensagem X
B(X)	Função de ofuscamento (*blind*) aplicada à mensagem X. Esta função é parametrizada por um valor secreto gerado pelo cliente e conhecido somente por ele.
C	Navegador do usuário (Cliente)
CP	Entidade ou endereço da entidade CP
CP_list	Lista de CPs aceitos por RP
IdP	Entidade ou endereço da entidade IdP
Info	Conjunto de atributos (informação) do usuário
$MAC_{IdP}(X)$	Código de autenticação da mensagem X. O MAC é gerado por IdP a partir de um segredo compartilhado com RP (conforme especificado pelo OpenID)
N_{RP}	*Nonce* (valor aleatório de uso único) gerado por RP
pID	Pseudônimo criado ou escolhido por C em IdP
pID_list	Lista com sugestões de pseudônimos para C (tipicamente, os pseudônimos criados em IdP pelo cliente em algum momento)
req_info	Atributos (informações) requisitados por RP para a autorização do cliente no serviço
RP	Entidade ou endereço da entidade RP
$Sign_{CP}(X)$	Assinatura de X por CP (tipicamente o *hash* da mensagem X encriptada com a chave privada de CP)
Tk	Mensagem (*token*) composta por pID, N_{RP}, info

A seguir, a Tabela 4 apresenta as mensagens do protocolo no modo de atributos múltiplos por credencial. Para efeito de ilustração, a autenticação entre o cliente e CP (cujo método deve ser especificado por CP) é representada por meio das mensagens M7 e M8; entretanto, cabe notar que, dependendo do método escolhido, a autenticação pode necessitar de mais de duas mensagens trocadas.

Tabela 4 - Mensagens do protocolo NibbleID no modo de atributos múltiplos por credencial

#	Direção	Conteúdo das mensagens trocadas em HTTPS
M1	C → RP	IdP
M2	RP → C	IdP, RP, N_{RP}, req_info, CP_list
M3	C → IdP	IdP, RP, N_{RP}, req_info, CP_list
M4	IdP → C	pID_list
M5	C → IdP	PID
M6	IdP → C	CP_list, pID, N_{RP}, req_info
M7	C → CP	Autenticação entre C e CP (Parâmetros)
M8	CP → C	Autenticação entre C e CP (OK)
M9	C → CP	B(pID, N_{RP}), info
M10	CP → C	$Sign_{CP}$(B(pID,N_{RP}), info)
M11	C → IdP	CP, info, $Sign_{CP}$(pID, N_{RP}, info)
M12	IdP → C	CP, RP, Tk, $Sign_{CP}$(Tk), MAC_{IdP}(RP, Tk, $Sign_{CP}$(Tk)) onde: Tk=pID, N_{RP}, info
M13	C → RP	CP, RP, Tk, $Sign_{CP}$(Tk), MAC_{IdP}(RP, Tk, $Sign_{CP}$(Tk))
M14	RP → C	OK

6.3.4 Mensagens do protocolo para o modo de atributos únicos por credencial

O formato das mensagens trocadas no protocolo para o modo de atributos únicos difere do modo de atributos múltiplos apenas nas mensagens M9 a M13. A Tabela 6 apresenta estas mensagens para o modo de atributos únicos por credencial. As novas notações introduzidas para esta descrição são apresentadas na Tabela 5.

Tabela 5 - Notação específica utilizada modo de atributos únicos por credencial

Notação	Significado
att	Atributo do usuário
Λ	Conjunto de atributos do usuário (att_1, att_2, ..., att_n)
att_sel	Atributo do usuário selecionado para a apresentação em RP
ϒ	Conjunto de atributos selecionados pelo usuário para a apresentação em um RP (att_sel_1, att_sel_2, ..., att_sel_m)

Tabela 6 - Mensagens do protocolo NibbleID no modo de atributos múltiplos por credencial

#	Direção	Conteúdo das mensagens trocadas em HTTPS
M9	C → CP	$B(pID, N_{RP})$
M10	CP → C	$Sign_{CP}(B(pID,N_{RP}), att_1)$, $Sign_{CP}(B(pID,N_{RP}), att_2)$, ..., $Sign_{CP}(B(pID,N_{RP}), att_n)$ onde: Λ = conjunto de atributos do usuário $\Lambda = \{att_1, att_2, ..., att_n,\}, n \in \mathbb{N}^*$
M11	C → IdP	CP, info, $Sign_{CP}(pID, N_{RP}, att\_sel_1)$, $Sign_{CP}(pID, N_{RP}, att\_sel_2)$, ..., $Sign_{CP}(pID, N_{RP}, att\_sel_m)$ onde: Υ= conjunto de atributos selecionados $\Upsilon = \{att\_sel_1, att\_sel_2, ..., att\_sel_m,\}, m \in \mathbb{N}^*$ $\Upsilon \subset \Lambda, m \leq n$
M12	IdP → C	CP, RP, $Tk_1, Tk_2, ..., Tk_m$, $Sign_{CP}(Tk_1)$, $Sign_{CP}(Tk_2)$, ..., $Sign_{CP}(Tk_m)$, $MAC_{IdP}(RP, Tk_1, Tk_2, ..., Tk_m, Sign_{CP}(Tk_1), Sign_{CP}(Tk_2), ..., Sign_{CP}(Tk_m))$ onde: $Tk_i = (pID, N_{RP}, att\_sel_i)$, $i \in \mathbb{N}^*, i \leq m$;
M13	C → RP	CP, RP, $Tk_1, Tk_2, ..., Tk_m$, $Sign_{CP}(Tk_1)$, $Sign_{CP}(Tk_2)$, ..., $Sign_{CP}(Tk_m)$, $MAC_{IdP}(RP, Tk_1, Tk_2, ..., Tk_m, Sign_{CP}(Tk_1), Sign_{CP}(Tk_2), ..., Sign_{CP}(Tk_m))$

6.3.5 Acessos subsequentes em um RP sem apresentação de credencial

Considere-se um usuário que comprova uma identidade parcial a um RP por meio de uma ou mais credenciais. Visto que uma identidade parcial é associada a um único

pseudônimo, a reapresentação deste pseudônimo pode, em princípio, ser utilizada para permitir acessos subsequentes desse usuário ao RP sem a reapresentação das credenciais. Em outras palavras, um usuário que comprove ser o dono de um pseudônimo pode, em princípio, se beneficiar da identidade parcial já confirmada para aquele pseudônimo em um determinado serviço.

Com isso, cabe ao RP a decisão de permitir ou não o acesso do usuário baseado nas informações de identidade parcial comprovadas em conexões passadas, para um determinado pseudônimo. Uma discussão sobre os aspectos de segurança deste e de outros tipos de acesso é apresentada a seguir.

6.4 Considerações sobre alguns aspectos de segurança no NibbleID

Esta seção apresenta algumas características e considerações conhecidas a respeito da segurança da solução proposta.

6.4.1 Transferência de credenciais entre usuários

Embora o protocolo NibbleID previna o reúso de credenciais entre usuários por meio da associação da credencial com pseudônimo e *nonce* (número aleatório de uso único gerado por RP) permanece ainda possível a transferência de credenciais no caso de um conluio entre usuários que se disponham a compartilhar entre si seus segredos (senhas) de acesso em IdP ou CP. Por exemplo, considere-se o seguinte cenário:

- Usuário U_A é membro em um emissor de credenciais CP_A
- Um serviço RP_A aceita as credenciais de CP_A (permite acesso aos membros de CP_A)
- Usuário U_B não é membro de CP_A, porém deseja acessar RP_A

Nesse cenário, caso U_A informe sua senha de acesso em CP_A para U_B, torna-se possível o acesso de U_B em RP_A de maneira anônima (por meio de um pseudônimo de U_B). Esse problema é também discutido em outras soluções, como o U-Prove e o Idemix e, embora não existam soluções definitivas para essa questão, diferentes abordagens são consideradas. Por exemplo, o U-Prove defende o desencorajamento no empréstimo de credenciais (*tokens* U-Prove) a terceiros por meio do armazenamento de informações sensíveis (o número do cartão de crédito, por exemplo) dentro da credencial do usuário. Essa informação sensível seria, então, acessível a qualquer outro usuário que recebesse a chave secreta referente a essa credencial do usuário original (**dono do *token* U-Prove**). O problema nessa abordagem é que não é possível garantir que o emissor de credenciais de fato conheça alguma informação sensível para ser armazenada na credencial do usuário. Além disso, o conceito de informação sensível para um usuário depende sempre da relação de confiança que os usuários possuem entre si. Por exemplo, uma informação que poderia ser considerada como sensível (como o número de cartão de crédito) pode não ser um segredo sensível entre membros de uma mesma família (relação forte de confiança). Logo, mesmo com o conhecimento de informações secretas de um usuário por parte do emissor de credenciais, o empréstimo de credenciais entre pessoas com uma forte relação de confiança continuaria possível.

Já o **Idemix** aborda esse problema utilizando a sua capacidade de "*desanonimizar*" uma credencial (***token* Idemix**). O Idemix provê um mecanismo de revelação da identidade original de um usuário caso exista a suspeita de mau uso das credenciais. Desta forma, a ideia de prevenção de transferência de credenciais baseia-se na punição de usuários descobertos realizando tal prática. No entanto, essa abordagem do Idemix também traz alguns problemas. Em primeiro lugar, o mecanismo de revelação de identidade em si é questionado por alguns autores (MOSTOWSKI; VULLERS, 2011) por ser contrário à própria ideia de anonimato fornecido em tais sistemas. Em segundo lugar, salvo alguns casos extremos, é bastante difícil comprovar uma suspeita de empréstimo de credenciais entre usuários no sistema, já que tais ações tendem a ser parecidas com o uso regular do sistema.

Outra abordagem possível em alguns cenários é o desencorajamento do empréstimo de credenciais por meio de imposição de limites na geração das credenciais. Por exemplo,

uma credencial pode ser emitida por um CP a um usuário caso este não tenha extrapolado um determinado limite definido por:

- Número de credenciais – Ex.: Cinco emissões de credenciais por semana
- Tipos de atributos – Ex.: Uma emissão de credencial com o atributo "possui direito a voto" a cada eleição
- Valor associado a um atributo – Ex.: Emissão de credenciais cujos valores associados ao atributo "horas de acesso" somados não ultrapassem 100 horas.

No NibbleID, a contabilização e imposição de limites na geração da credencial podem ser feitas pelo CP durante a etapa 8 do protocolo, descrito na Figura 15.

Com essa discussão, é possível afirmar que o problema de transferência deliberada de credenciais entre usuários pode ser minimizado com a aplicação de políticas de acesso aliadas à contabilização na geração de credenciais ou com a existência de condições específicas (como o da existência de informações sensíveis escondidas na credencial ou possibilidade de investigação de fraudes). Porém, de um ponto de vista mais pragmático, pode-se dizer que em SGIs com suporte à privacidade (incluindo-se o NibbleID e os exemplos citados, U-Prove e Idemix) não existem soluções definitivas para evitar este problema, quando usuários decidem entregar suas senhas secretas ou permitir acesso a outro usuário. Algo similar pode ser observado para os casos de coerção, onde um usuário é forçado a fornecer ou dar acesso a outro usuário por intimidação.

6.4.2 Política de acessos subsequentes a um RP

Como foi visto anteriormente, é possível o acesso subsequente a um RP por meio de um pseudônimo sem a reapresentação de credenciais. A permissão desse tipo de acesso possibilita uma maior agilidade no processo de autorização em um RP, uma vez que neste caso apenas a comprovação prevista pelo mecanismo do OpenID se faz necessária. Por outro lado, essa permissão não fornece garantias ao RP de que as informações apresentadas para o pseudônimo (usuário) em um momento anterior **permanecem** ainda **válidas** (informações de credenciais antigas podem ter sido alteradas em CP).

Com isso, para um maior rigor no controle de acesso, pode ser interessante para RP estipular uma políticas que obriguem o usuário a sempre prover as credenciais em qualquer acesso ao serviço. Isso garante ao RP que a associação do pseudônimo com a credencial seja atual. Outra opção de política é a aceitação de tais acessos (sem a reapresentação de credenciais) até um determinado limite ou tempo máximo. Após esse limite o usuário é obrigado a reapresentar as credenciais de modo a **renovar** a associação entre o pseudônimo e a identidade parcial.

6.4.3 Escolha de um IdP pelo usuário

O IdP na arquitetura do NibbleID (assim como o IdP na arquitetura OpenID) pode ser escolhido livremente pelo usuário. É importante, porém, ressaltar que, como mantenedor da pseudo-identidade do usuário (identidade parcial por meio do pseudônimo), o IdP deve ser confiável do ponto de vista do usuário. Um IdP mal-intencionado pode, em benefício próprio, apresentar pseudônimos em RPs no lugar do usuário real onde a reapresentação da credencial não é necessária. Ainda, um IdP malicioso poderia entregar para o usuário um *software* malicioso[19] (na etapa 5 do protocolo, descrito na Figura 15) que remove a condição de não-relacionabilidade adicionada pelo mecanismo de assinatura cega. O *software* poderia, por exemplo, enviar informações pessoais do usuário para o IdP.

A fim de minimizar problemas como esses, IdPs considerados como não confiáveis devem ser adicionados a uma lista negra de modo a não serem aceitos pelos RPs ou pelos usuários.

[19] Nesse caso, a opção da pré-instalação de um plug-in de uma fonte confiável seria uma opção mais segura para o usuário.

6.4.4 Questões de segurança referentes ao OpenID

Visto que o NibbleID utiliza o OpenID como mecanismo de *Single Sign On* e provimento de pseudônimos, é natural que as considerações a respeito da segurança no OpenID sejam também observadas no NibbleID. Por exemplo, no OpenID é recomendável que o OP (IdP no NibbleID) possa alertar, ou mesmo impedir, o acesso de um usuário a um RP cuja comunicação não esteja protegida por HTTPS. Portanto, esta mesma recomendação é também aconselhável para o NibbleID*.*

Uma análise da segurança no OpenID é provida em (DOMINICINI *et al.*, 2010), de co-autoria com o autor desse trabalho. Mais informações sobre recomendações de segurança para o OpenID podem ser também encontradas em (RECORDON; FITZPATRICK, 2007).

6.5 Resumo do Capítulo

Esse capítulo apresentou o NibbleID como uma proposta de sistema para gerenciamento de identidades na Web com suporte à privacidade do usuário por meio de credenciais. Por meio desse sistema é possível a apresentação de uma identidade parcial cujos atributos foram escolhidos pelo usuário para acesso em um serviço na Web. A apresentação de tal identidade parcial ocorre de modo que o vazamento de informações do usuário (não consentidas) para o serviço é evitado.

A arquitetura do NibbleID inova ao propor papéis distintos às entidades CP e IdP. Essa arquitetura se adequa melhor às possíveis relações de confiança existentes entre usuários e provedores (sejam eles provedores de identidade, de credenciais ou de serviços) na Web e possibilita uma maior flexibilidade na aplicação da solução. Por exemplo, a solução possibilita que um usuário gerencie identidades parciais compostas por atributos provenientes de diferentes organizações (CPs) em um provedor de identidades (IdP) centralizado de sua escolha.

O protocolo NibbleID também traz flexibilidade ao permitir que diferentes níveis de segurança ou privacidade sejam adotados de acordo com as políticas dos provedores de serviço (RPs) ou dos provedores de credenciais (CPs). Ainda, o protocolo traz flexibilidade para o usuário, de modo a permitir que o mesmo decida por diferentes tipos de pseudônimos que possibilitam diferentes condições para a preservação de sua privacidade.

Esse capítulo apresentou também uma discussão crítica a respeito de aspectos de segurança do protocolo e expôs algumas de suas limitações, por exemplo, na prevenção da transferência de credenciais entre usuários em conluio. Outras análises serão apresentadas no capítulo 8.

No capítulo a seguir, será apresentada uma análise formal do protocolo NibbleID em lógica BAN (BURROWS; ABADI; NEEDHAM, 1989). Essa análise permite a verificação do cumprimento dos objetivos do protocolo no que diz respeito à autenticação de mensagens e entidades do sistema. Por meio da lógica BAN será possível, também a análise das informações vistas em cada uma das entidades durante a execução do protocolo. Esta última é importante para verificar que informações sensíveis (i.e.: informações não aprovadas pelo usuário ou que o associem diretamente a um pseudônimo) não sejam indevidamente expostas entre as entidades do sistema pelo protocolo.

7 ANÁLISE FORMAL DO PROTOCOLO NIBBLEID

O capítulo anterior apresentou o protocolo NibbleID. De maneira geral, o objetivo deste protocolo é permitir que um **usuário** possa comprovar atributos selecionados de sua identidade a um **serviço** com a preservação de sua privacidade.

Com isso, este capítulo busca analisar o cumprimento desse objetivo por meio da análise formal do protocolo utilizando-se a lógica BAN (BURROWS; ABADI; NEEDHAM, 1989). Para facilitar a verificação do atendimento a este objetivo, esta análise foi dividida em duas partes distintas. Em primeiro lugar, deseja-se analisar se os **objetivos do protocolo na comprovação (autenticação) dos atributos** para um serviço ocorrem de maneira bem-sucedida. Essa primeira parte pode ser obtida por meio da aplicação direta da lógica BAN, conforme descrita por seus autores e a ser apresentada neste capítulo. Em segundo lugar, deseja-se demonstrar que a **privacidade do usuário é preservada no protocolo por meio da ocultação de informações** não consentidas pelo usuário e que possam ser utilizadas para sua identificação. Para isto, a lógica BAN será utilizada de modo a demonstrar que as informações obtidas pelas entidades do sistema devido à execução do protocolo não permitem a associação direta entre o pseudônimo e a identidade real do usuário.

7.1 Lógica BAN

Métodos formais são úteis para a análise de protocolos criptográficos e de autenticação. Dentre os métodos formais existentes, a lógica BAN proposta por **B**urrow, **A**badi e **N**eedham (BURROWS; ABADI; NEEDHAM, 1989) destaca-se como um dos métodos mais utilizado em tais análises e na literatura (MENEZES; OORSCHOT, VAN; VANSTONE, 2001). A lógica BAN foi escolhida neste trabalho devido à sua grande adoção na análise de protocolos de autenticação e constante aperfeiçoamento em publicações especializadas na área (ABDELMAJID *et al.*, 2010).

Por meio da lógica BAN, é possível demonstrar que os objetivos de um protocolo de autenticação ocorrem de maneira bem-sucedida. Além disso, essa lógica possibilita a

descoberta de falhas e redundâncias no protocolo de segurança (MENEZES; OORSCHOT, VAN; VANSTONE, 2001). De fato, de acordo com (SYVERSON, PAUL; CERVESATO, 2001), a lógica BAN tem sido útil no descobrimento de falhas em diversos protocolos de segurança já publicados.

De maneira geral, a análise com a lógica BAN é baseada em derivações das mensagens do protocolo e das premissas iniciais do sistema analisado. Essas derivações ocorrem por meio da aplicação de expressões lógicas (postulados ou regras) definidos por BAN. Com tais derivações, é possível analisar o conhecimento ou crença de cada entidade do sistema em cada passo do protocolo. As derivações permitem também a demonstração de que os objetivos de um protocolo de autenticação são cumpridos durante sua execução. Por exemplo, pode-se demonstrar, que em um protocolo, duas entidades comprovam a existência de um segredo compartilhado entre elas para fins de autenticação mútua. Ainda, pode-se demonstrar que uma entidade é atestada ou autenticada por uma terceira entidade confiável do sistema.

A Tabela 7, a seguir, apresenta as **expressões** utilizadas na Lógica BAN e os seus respectivos significados. Os símbolos **P** e **Q** representam **entidades** do sistema analisado, tais como computadores, serviços ou pessoas. Os símbolos **X** e **Y**, por sua vez, representam **expressões** BAN ou **mensagens** do protocolo. Já o símbolo **K** é utilizado para denotar chaves criptográficas em geral, enquanto $\mathbf{K^{-1}}$ é utilizado particularmente para indicar chaves privadas. Por fim, o símbolo **B** foi inserido nas expressões da lógica BAN original por este trabalho para representar a função de ofuscamento (conforme definido na seção 3.3.1) de mensagens e permitir a modelagem das assinaturas cegas utilizadas no NibbleID.

Tabela 7 – Expressões da lógica BAN

Expressão	Significado
$P \models X$	P acredita que X é verdadeiro
$P \triangleleft X$	P enxerga X (P recebeu X em algum momento).
$P \mid\sim X$	P enviou X em algum momento. Essa expressão implica também que P enxerga X no momento do envio.
$P \Rightarrow X$	P tem jurisdição sobre X. Isto significa que a entidade P possui autoridade sobre X para a sua criação e deve ser considerada como confiável para tal.
$\#(X)$	X é recente (*fresh*) e não foi utilizado em nenhum momento anterior à execução do protocolo. Tipicamente, X é um valor aleatório.
$P \overset{K}{\leftrightarrow} Q$	P e Q possuem uma chave compartilhada K que pode ser utilizada para a comunicação segura entre eles. A chave K é conhecida somente por P e Q. Opcionalmente, K pode ser conhecido também por entidades da confiança de P e Q.
$\overset{K}{\rightarrow} P$	P possui uma chave pública K. Isto implica também que a chave privada relacionada, K^{-1}, é conhecida apenas por P. Opcionalmente, K^{-1} pode ser conhecido também por entidades da confiança de P.
$P \overset{X}{\Leftrightarrow} Q$	P e Q possuem um segredo compartilhado X. O segredo X é conhecido somente por P e Q. Opcionalmente, X pode ser conhecido também por entidades da confiança de P e Q.
$\{X\}_K$	X está encriptado com a chave K.
$\langle X \rangle_Y$	X é combinado com Y. De acordo com BAN, Y é geralmente um segredo o qual pode ser utilizado para comprovar a identidade do gerador da mensagem X.
$B_P^K(X)$	X foi ofuscado por P com a finalidade de ser assinado por meio do mecanismo de assinaturas cegas. A função inversa B^{-1} é conhecida somente por P. O índice K indica a chave pública do emissor de assinaturas cegas (parâmetro necessário para a geração de B)

7.1.1 Postulados da lógica BAN

A lógica BAN possui diversos postulados ou regras utilizados para a derivação de mensagens e premissas do protocolo. Estes postulados ou regras seguem preceitos tipicamente estudados em criptografia e protocolos de segurança. Por exemplo, o postulado: $\frac{P \models Q \overset{K}{\leftrightarrow} P \ , P \triangleleft \{X\}_K}{P \triangleleft X}$ indica que **se** a entidade P acredita na existência de uma chave K compartilhada entre P e Q (isto implica que P possui a chave K) **e** uma vez recebeu ou viu uma mensagem cifrada com K, **logo**, é possível concluir que P é capaz de ver o conteúdo X da mensagem cifrada, uma vez que é possível para P decriptar $\{X\}_k$.

Mais informações sobre a lógica BAN podem ser encontradas em (BURROWS; ABADI; NEEDHAM, 1989), (MENEZES; OORSCHOT, VAN; VANSTONE, 2001) e (SYVERSON, PAUL; CERVESATO, 2001).

A seguir, são apresentadas as regras da lógica BAN utilizadas nesse trabalho:

- Regra1 (Remetente da mensagem - chaves compartilhadas):

$$\frac{P \models Q \overset{K}{\leftrightarrow} P \ , \ P \triangleleft \{X\}_K}{P \models Q \mid\sim X} \qquad (R1)$$

- Regra R2 (Remetente da mensagem -chaves públicas):

$$\frac{P \models \overset{K}{\rightarrow} Q \ , \ P \triangleleft \{X\}_{K^{-1}}}{P \models Q \mid\sim X} \qquad (R2)$$

- Regra R3: (Crença por mensagem recente):

$$\frac{P \models \#(X) \ , \ P \models Q \mid\sim X}{P \models Q \models X} \qquad (R3)$$

- Regra R4 (Jurisdição):

$$\frac{P \models Q \Rightarrow X,\ P \models Q \models X}{P \models X} \quad (R4)$$

- Regra R5 (Visualização de componentes da mensagem):

$$\frac{P \lhd (X, Y)}{P \lhd X} \quad (R5a)$$

$$\frac{P \models \overset{K}{\rightarrow} Q,\ P \lhd \{X\}_{K^{-1}}}{P \lhd X} \quad (R5b)$$

$$\frac{P \models \overset{K}{\rightarrow} P,\ P \lhd \{X\}_{K}}{P \lhd X} \quad (R5c)$$

$$\frac{P \models Q \overset{K}{\leftrightarrow} P\ ,\ P \lhd \{X\}_{K}}{P \lhd X} \quad (R5d)$$

- Regra R6 (Verificação de mensagem recente por componente):

$$\frac{P \models \#(X)}{P \models \#(X, Y)} \quad (R6)$$

- Regra 7 (Propriedade do operador "$\models$"):

$$\frac{P \models X, P \models Y}{P \models (X, Y)} \quad (R7a)$$

$$\frac{P \models (X, Y)}{P \models (X)} \quad (R7b)$$

$$\frac{P \models Q \models (X, Y)}{P \models Q \models X} \quad (R7c)$$

- Regra 8 (Propriedade do operador "$\vdash\sim$"):

$$\frac{P \vdash\sim (X, Y)}{P \vdash\sim (X)} \quad (R8)$$

- Regra 9 (Desofuscamento de assinatura):

$$\frac{P \triangleright B_P^K(X)\ ,\ P \lhd \{B_P^K(X), Y\}_{K^{-1}}\ ,\ P \equiv \overset{K}{\rightarrow} Q}{P \lhd \{X, Y\}_{K^{-1}}} \quad (R9)$$

7.2 Análise do formal do protocolo NibbleID na lógica BAN

A análise do protocolo NibbleID neste trabalho ocorre por meio de quatro passos, apresentados a seguir:

- 1º passo: Descrição das **premissas iniciais** do protocolo em notação BAN;
- 2º passo: Enumeração dos **objetivos de autenticação** do protocolo em notação BAN;
- 3º passo: Descrição das **mensagens do protocolo** segundo o modelo **idealizado** previsto por BAN;
- 4º passo: **Derivação das expressões** obtidas no primeiro e terceiro passos por meio dos postulados BAN. Por meio dessas derivações, deve ser possível a **obtenção** das expressões enumeradas nos **objetivos de autenticação**, descritos no 3º passo.

Nesta análise, o quarto passo será também utilizado para demonstrar formalmente quais informações a respeito do **usuário** podem ser obtidas em cada uma das entidades durante a execução do protocolo.
Seguindo estes passos, as seções a seguir demonstram a análise do protocolo NibbleID para o modo de atributos **múltiplos** por credencial. Posteriormente, na seção 7.4, serão observadas quais alterações se aplicam nesta análise para o modo de atributos **únicos**.

7.2.1 Premissas Iniciais

Como primeiro passo da análise, são descritas as premissas iniciais do protocolo NibbleID para o **modo de atributos múltiplos** por credencial utilizando-se a notação BAN.

A Tabela 3, a seguir, apresenta as notações utilizadas neste capítulo. A maior parte da notação apresentada nesta tabela foi reaproveitada do capítulo anterior.

Tabela 8 - Notações utilizadas para a descrição das mensagens na lógica BAN

Notação	Significado
C	Navegador Web do usuário (Cliente)
CP	Entidade ou endereço da entidade CP
CP_list	Lista de CPs aceitos por RP
IdP	Entidade ou endereço da entidade IdP
Info	Conjunto de atributos (informação) do usuário
N_{RP}	*Nonce* (valor aleatório de uso único) gerado por RP
pID	Pseudônimo criado ou escolhido por C em IdP
pID_list	Lista com sugestões de pseudônimos para C (tipicamente, os pseudônimos criados em IdP pelo cliente em algum momento)
req_info	Atributos (informações) requisitados por RP para a autorização do cliente no serviço
RP	Entidade ou endereço da entidade RP
Tk	Mensagem (*token*) composta por pID, N_{RP}, info

A seguir, as premissas P1 a P12 descrevem as chaves compartilhadas de sessão obtidas ao se estabelecer um canal HTTPS seguro entre o navegador do **usuário** e as **entidades RP (**premissas P1 a P4), IdP (premissas P5 a P8) e CP (premissas P9 a P12). Para simplificação da análise, consideram-se as sessões HTTPS estabelecidas imediatamente antes do início do protocolo. Isso evita que o procedimento de estabelecimento da sessão SSL/TLS seja também analisado pela lógica BAN.

P1. $C \models C \overset{K_{C-RP}}{\longleftrightarrow} RP$

P2. $RP \models C \overset{K_{C-RP}}{\longleftrightarrow} RP$

P3. $C \models \#(C \overset{K_{C-RP}}{\longleftrightarrow} RP)$

P4. $RP \models \#(C \overset{K_{C-RP}}{\longleftrightarrow} RP)$

P5. $C \models C \overset{K_{C-IdP}}{\longleftrightarrow} IdP$

P6. $IdP \models C \overset{K_{C-IdP}}{\longleftrightarrow} IdP$

P7. $C \models \#(C \overset{K_{C-IdP}}{\longleftrightarrow} IdP)$

P8. $IdP \models \#(C \overset{K_{C-IdP}}{\longleftrightarrow} IdP)$

P9. $C \models C \overset{K_{C-CP}}{\longleftrightarrow} CP$

P10. $CP \models C \overset{K_{C-CP}}{\longleftrightarrow} CP$

P11. $C \models \#(C \overset{K_{C-CP}}{\longleftrightarrow} CP)$

P12. $CP \models \#(C \overset{K_{C-CP}}{\longleftrightarrow} CP)$

As premissas P13 e P14 descrevem o segredo compartilhado entre RP e IdP, utilizado para a criação do código de autenticação de mensagem (*MAC - Message Autentication Code*) das asserções OpenID[20]. O estabelecimento desse segredo compartilhado foi mencionado na descrição do OpenID, na seção 2.4.1.

P13. $RP \models RP \overset{K_{RP-IdP}}{\longleftrightarrow} IdP$

P14. $IdP \models RP \overset{K_{RP-IdP}}{\longleftrightarrow} IdP$

As premissas P15 e P16 indicam o conhecimento e crença de RP e C na chave pública de CP (entidade confiável para a geração de credenciais).

P15. $RP \models \overset{K_{CP}}{\longrightarrow} CP$

P16. $C \models \overset{K_{CP}}{\longrightarrow} CP$

[20] Embora o OpenID preveja outro método para a verificação do MAC, conforme (RECORDON; FITZPATRICK, 2007), o resultado final da análise do protocolo NibbleID permanece o mesmo para qualquer um dos métodos utilizados.

Uma vez que no protocolo o valor do *nonce* N_{RP} é gerado aleatoriamente por RP, é correto presumir que essa entidade acredita que N_{RP} é uma nova mensagem (*fresh*). A premissa P17 indica essa condição.

P17. $RP \models \#(N_{RP})$

A premissa P18 indica que C é o emissor da função de assinatura parcialmente cega. A chave K_{CP} indica a chave pública de CP (destinatário da mensagem ofuscada), conhecida por C (como definido em P16).

P18. $C \mapsto B_C^{K_{CP}}$

Embora o método de autenticação utilizado entre C e CP não faça parte do escopo do protocolo NibbleID, é necessária a adoção de algum método para que os objetivos do protocolo sejam alcançados na análise BAN (basicamente, é necessário que CP autentique C para o provimento da credencial). Portanto, as premissas P19 a P20 descrevem uma condição onde há um segredo compartilhado (uma senha, por exemplo) entre estas entidades. Já em P19 e P20, o elemento Ts é utilizado como um *nonce* para a mensagem de autenticação (por exemplo, o tempo de relógio – *timestamp* – no caso de sincronismo entre as entidades).

P19. $C \models C \overset{pw}{\Leftrightarrow} CP$

P20. $CP \models C \overset{pw}{\Leftrightarrow} CP$

P21. $C \models \#(Ts)$

P22. $CP \models \#(Ts)$

As três últimas premissas, P23 a P25, indicam que as entidades são capazes de visualizar seus próprios endereços. A definição destas premissas será útil na análise das informações conhecidas pela entidade ao final da execução do protocolo.

P23. $RP \triangleleft RP$

P24. $IdP \triangleleft IdP$

P25. $CP \triangleleft CP$

A seguir, serão descritos os objetivos do protocolo de acordo com a notação BAN.

7.2.2 Objetivos do protocolo para a autenticação

O segundo passo na análise BAN neste trabalho é a descrição dos objetivos de autenticação do protocolo. Estes objetivos são expressos em notação BAN a seguir:

O1. $RP \models CP \models Tk$

O2. $RP \models IdP \models (RP, pID, N_{RP})$

O3. $RP \models C \sim Tk$

Essas expressões são obtidas pela análise das condições a serem verificadas por RP para a permissão no acesso ao serviço. No caso, RP deve ser capaz de verificar as seguintes condições ao término da execução do protocolo:

- **Condição 1:** CP autenticou a mensagem Tk (composta por pID, N_{RP} e info). Em lógica BAN, **CP** $\models$ **Tk** (CP acredita em Tk).
- Esta condição garante à RP que CP verificou que **info** é verdadeiro para o usuário que se apresenta pelo pseudônimo **pID**. O nonce $\mathbf{N_{RP}}$ gerado por RP garante que a mensagem é recente (*fresh*).
- **Condição 2:** IdP autenticou a mensagem RP, pID e N_{RP}. Em lógica BAN, **IdP** $\models$ **(RP, pID,** $\mathbf{N_{RP}}$**)**.
- Esta condição garante a RP que a asserção OpenID sobre o pseudônimo é válida (reconhecida por IdP). O campo **RP** (tipicamente contendo a URL de RP) é utilizado para evitar que um RP malicioso atue como *man-in-the-middle*[21] entre o usuário e um RP legítimo.
- **Condição 3:** O cliente[22] C foi o elemento responsável pela obtenção de Tk. Em lógica BAN, **C** $\sim$ **Tk**.
- Esta condição garante a RP que C obteve a mensagem Tk (baseado no *nonce* N_{RP}) com êxito.

[21] Um RP malicioso poderia atuar como um intermediário entre um RP legítimo e um usuário na tentativa de obter acesso indevido ao serviço por meio das credenciais do cliente.

[22] Nesse momento da autenticação, o cliente é visto como um navegador Web cuja sessão permanece ativa no servidor Web em RP.

A seguir, será apresentada a versão do protocolo do NibbleID, reescrito de acordo com o modelo idealizado segundo BAN.

7.2.3 Protocolo Idealizado segundo BAN

O terceiro passo na análise BAN deste trabalho é a descrição do protocolo no modo chamado "idealizado" (BURROWS; ABADI; NEEDHAM, 1989), (SYVERSON, PAUL; CERVESATO, 2001). Neste modo, as mensagens do protocolo são reescritas de modo a remover mensagens de protocolo ou componentes que não sejam úteis para a análise. Por exemplo, informações duplicadas ou que não contribuem para o aumento de crenças das entidades devem ser removidas. Esse modo permite também que simplificações sejam feitas caso não alterem o propósito de uma mensagem de protocolo ou a crença das entidades. Por exemplo, considere-se a assinatura de uma mensagem obtida por meio da encriptação de seu resumo criptográfico (*hash*)[23] por meio da chave privada do assinante. Em uma simplificação para a análise BAN, essa assinatura pode ser vista como a encriptação direta da mensagem original sem prejuízos para a análise na maioria dos casos (tipicamente se a mensagem original acompanhar a sua assinatura digital).

A Tabela 10, adiante, apresenta o protocolo NibbleID idealizado segundo BAN para o modo de atributos múltiplos por credencial. Esta tabela é baseada nas mensagens apresentadas na Tabela 4, no capítulo 6. A fim de facilitar a leitura do texto, esta tabela é reapresentada a seguir, na forma da Tabela 9.

[23] A encriptação do *hash* da mensagem (ao oposto da mensagem completa) para a geração de uma assinatura permite melhor desempenho e uma assinatura de tamanho fixo tipicamente menor que a mensagem original.

Tabela 9 – Reapresentação da Tabela 4 (Mensagens do protocolo NibbleID no modo de atributos múltiplos por credencial)

#	Direção	Conteúdo das mensagens trocadas em HTTPS
M1	C → RP	IdP
M2	RP → C	IdP, RP, N_{RP}, req_info, CP_list
M3	C → IdP	IdP, RP, N_{RP}, req_info, CP_list
M4	IdP → C	pID_list
M5	C → IdP	pID
M6	IdP → C	CP_list, pID, N_{RP}, req_info
M7	C → CP	Autenticação entre C e CP (Parâmetros)
M8	CP → C	Autenticação entre C e CP (OK)
M9	C → CP	B(pID, N_{RP}), info
M10	CP → C	$Sign_{CP}$(B(pID,N_{RP}), info)
M11	C → IdP	CP, info, $Sign_{CP}$(pID, N_{RP}, info)
M12	IdP → C	CP, RP, Tk, $Sign_{CP}$(Tk), MAC_{IdP}(RP, Tk, $Sign_{CP}$(Tk)) onde: Tk=pID, N_{RP}, info
M13	C → RP	CP, RP, Tk, $Sign_{CP}$(Tk), MAC_{IdP}(RP, Tk, $Sign_{CP}$(Tk))
M14	RP → C	OK

Tabela 10 - Protocolo NibbleID idealizado segundo BAN (modo de atributos múltiplos por credencial)

#	Direção	Mensagens idealizadas segundo BAN
M1	C → RP	$\{\mathrm{IdP}\}_{K_{C-RP}}$
M2	RP → C	$\{\mathrm{IdP}, \mathrm{RP}, N_{RP}, \mathrm{req\_info}, \mathrm{CP\_list}\}_{K_{C-RP}}$
M3	C → IdP	$\{\mathrm{IdP}, \mathrm{RP}, N_{RP}, \mathrm{req\_info}, \mathrm{CP\_list}\}_{K_{C-IdP}}$
M4	IdP → C	$\{\mathrm{pID\_list}\}_{K_{C-IdP}}$
M5	C → IdP	$\{\mathrm{pID}\}_{K_{C-IdP}}$
M6	IdP → C	$\{\mathrm{CP\_list}, \mathrm{pID}, N_{RP}, \mathrm{req\_info}\}_{K_{C-IdP}}$
M7	C → CP	$\{\langle \mathrm{username}, T_S \rangle_{pw}\}_{K_{C-CP}}$
M9	C → CP	$\{B_C^{K_{CP}}(\mathrm{pID}, N_{RP}), \mathrm{info}\}_{K_{C-CP}}$
M10	CP → C	$\{\{B_C^{K_{CP}}(\mathrm{pID}, N_{RP}), \mathrm{info}\}_{K_{CP}^{-1}}\}_{K_{C-CP}}$
M11	C → IdP	$\{\mathrm{CP}, \mathrm{info}, \{\mathrm{pID}, N_{RP}, \mathrm{info}\}_{K_{CP}^{-1}}\}_{K_{C-IdP}}$
M12	IdP → C	$\{\mathrm{CP}, \mathrm{RP}, \mathrm{Tk}, \{\mathrm{Tk}\}_{K_{CP}^{-1}}, \{\mathrm{RP}, \mathrm{Tk}, \{\mathrm{Tk}\}_{K_{CP}^{-1}}\}_{K_{RP-IdP}}\}_{K_{C-IdP}}$ onde Tk=(pID,N_{RP},info)
M13	C → RP	$\{\mathrm{CP}, \mathrm{RP}, \mathrm{Tk}, \{\mathrm{Tk}\}_{K_{CP}^{-1}}, \{\mathrm{RP}, \mathrm{Tk}, \{\mathrm{Tk}\}_{K_{CP}^{-1}}\}_{K_{RP-IdP}}\}_{K_{C-RP}}$

No protocolo NibbleID idealizado segundo BAN, apresentado na Tabela 10, as mensagens M8 e M14 foram removidas por não contribuírem para a análise do protocolo. Ainda, ocorreu a simplificação na representação de assinaturas digitais por meio da encriptação direta da mensagem a ser assinada, conforme discutido no início desta seção. Esta mesma simplificação foi também aplicada para a geração do *MAC*, utilizado pela asserção OpenID.

Neste ponto, é importante enfatizar que, em uma análise BAN típica, todos os campos de mensagens do protocolo, que não contribuem para o processo de autenticação, são removidos no protocolo idealizado. Na análise presente, porém, optou-se por não remover tais informações[24], uma vez que um dos objetivos deste capítulo é analisar quais informações a respeito do usuário puderam ser obtidas em cada entidade ao fim da execução do protocolo.

A seguir, serão apresentadas as derivações obtidas por meio das mensagens descritas para o modelo idealizado BAN.

7.2.4 Derivações das expressões por meio dos postulados BAN

Como quarto passo da análise BAN deste trabalho, esta seção apresenta as derivações das expressões obtidas no modelo idealizado e na descrição das premissas iniciais do protocolo. Essas derivações ocorrem por meio da aplicação dos postulados BAN definidos na seção 7.1.1.

Por meio dessas derivações, deve ser possível:

- A obtenção das expressões enumeradas nos objetivos de autenticação;
- A análise de quais informações são obtidas (enxergadas $\triangleleft$) pelas entidades do sistema durante a execução do protocolo.

O procedimento de derivação ocorre para cada uma das mensagens do protocolo idealizado. Estas derivações são apresentadas a seguir.

Derivações da mensagem M1:

D1. $C \mid\sim \{IdP\}_{K_{C-RP}}$ (por M1)

D2. $C \triangleleft \{IdP\}_{K_{C-RP}}$ (Pelo significado de $\mid\sim$ em D1)

[24] Por exemplo, campos como *info, CP_list* e *req_info* poderiam ser removidos em uma análise BAN típica.

D3. $C \triangleleft IdP$ (por D2 e P1 em R5d)

D4. $RP \triangleleft \{IdP\}_{K_{C-RP}}$ (RP recebe M1)

D5. $RP \triangleleft IdP$ (D4 e P2 em R5d)

Derivações da mensagem M2:

D6. $RP \mathrel{|\sim} \{IdP, RP, N_{RP}, req\_info, CP\_list\}_{K_{C-RP}}$ (por M2)

D7. $RP \triangleleft \{IdP, RP, N_{RP}, req\_info, CP\_list\}_{K_{C-RP}}$ (Pelo significado de $\mathrel{|\sim}$ em D6)

D8. $RP \triangleleft N_{RP}$ (por D7 e P2 em R5d e R5a)

D9. $RP \triangleleft req\_info$ (por D7 e P2 em R5d e R5a)

D10. $RP \triangleleft CP\_list$ (por D7 e P2 em R5d e R5a)

D11. $C \triangleleft \{IdP, RP, N_{RP}, req\_info, CP\_list\}_{K_{C-RP}}$ (C recebe M2)

D12. $C \equiv RP \mathrel{|\sim} IdP, RP, N_{RP}, req\_info, CP\_list$ (D10 e P1 em R1)

D13. $C \triangleleft N_{RP}$ (D10 e P1 em R5d e R5a)

D14. $C \triangleleft req\_info$ (D10 e P1 em R5d e R5a)

D15. $C \triangleleft CP\_list$ (D10 e P1 em R5d e R5a)

D16. $C \triangleleft RP$ (D10 e P1 em R5d e R5a)

Derivações da mensagem M3 (redirecionamento):

D17. $IdP \triangleleft \{IdP, RP, N_{RP}, req\_info, CP\_list\}_{K_{C-IdP}}$ (IdP recebe M3)

D18. $IdP \equiv C \mathrel{|\sim} IdP, RP, N_{RP}, req\_info, CP\_list$ (D17 e P6 em R1)

D19. $IdP \triangleleft req\_info$ (D17 e P6 em R5d e R5a)

D20. $IdP \triangleleft N_{RP}$ (D17 e P6 em R5d e R5a)

D21. $IdP \triangleleft CP\_list$ (D17 e P6 em R5d e R5a)

D22. $IdP \triangleleft RP$ (D17 e P6 em R5d e R5a)

Derivações da mensagem M4:

D23. $\text{IdP} \mathrel{|\sim} \{\text{pID\_list}\}_{K_{C-IdP}}$ (por M4)

D24. $\text{IdP} \triangleleft \{\text{pID\_list}\}_{K_{C-IdP}}$ (Pelo significado de $\mathrel{|\sim}$ em D23)

D25. $\text{IdP} \triangleleft \text{pID\_list}$ (por D24 e P6 em R5d e R5a)

D26. $\text{C} \triangleleft \{\text{pID\_list}\}_{K_{C-IdP}}$ (C recebe M4)

D27. $\text{C} \equiv \text{IdP} \mathrel{|\sim} \text{pID\_list}$ (D26 e P5 em R1)

D28. $\text{C} \triangleleft \text{pID\_list}$ (D26 e P5 em R5d e R5a)

Derivações da mensagem M5:

D29. $\text{C} \mathrel{|\sim} \{\text{pID}\}_{K_{C-IdP}}$ (por M5)

D30. $\text{C} \triangleleft \{\text{pID}\}_{K_{C-IdP}}$ (Pelo significado de $\mathrel{|\sim}$ em D29)

D31. $\text{C} \triangleleft \text{pID}$ (D30 e P5 em R5d e R5a)

D32. $\text{IdP} \triangleleft \{\text{pID}\}_{K_{C-IdP}}$ (IdP recebe M5)

D33. $\text{IdP} \equiv \text{C} \mathrel{|\sim} \text{pID}$ (D32 e P6 em R1)

D34. $\text{IdP} \triangleleft \text{pID}$ (D32 e P6 em R5d e R5a)

Derivações da mensagem M6:

D35. $\text{C} \triangleleft \{\text{CP\_list}, \text{pID}, N_{RP}, \text{req\_info}\}_{K_{C-IdP}}$ (C recebe M6)

D36. $\text{C} \equiv \text{IdP} \mathrel{|\sim} \text{CP\_list}, \text{pID}, N_{RP}, \text{req\_info}$ (D35 e P5 em R1)

Derivações da mensagem M7:

D37. $\text{C} \mathrel{|\sim} \{\langle \text{username}, T_S \rangle_{pw}\}_{K_{C-CP}}$ (por M7)

D38. $\text{C} \triangleleft \{\langle \text{username}, T_S \rangle_{pw}\}_{K_{C-CP}}$ (Pelo significado de $\mathrel{|\sim}$ em D37)

D39. $\text{C} \triangleleft \text{username}$ (D38 e P9 em R5d e R5a)

D40. $C \triangleleft T_S$ (D38 e P9 em R5d e R5a)

D41. $C \triangleleft pw$ (por P19)

D42. $CP \triangleleft \{\langle username, T_S\rangle_{pw}\}_{K_{C-CP}}$ (CP recebe M7)

D43. $CP \equiv C \mid\sim \langle username, T_S\rangle_{pw}$ (D42 e P10 em R1)

D44. $CP \triangleleft \langle username, T_S\rangle_{pw}$ (D42 e P10 em R5d e R5a)

D45. $CP \triangleleft username$ (D42 e P10 em R5d e R5a)

D46. $CP \triangleleft T_S$ (D42 e P10 em R5d e R5a)

D47. $CP \triangleleft pw$ (por P20)

Derivações da mensagem M9:

D48. $C \mid\sim \{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{C-CP}}$ (por M9)

D49. $C \triangleleft \{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{C-CP}}$ (Pelo significado de $\mid\sim$ em D48)

D50. $C \triangleleft info$ (D49 e P9 em R5d e R5a)

D51. $C \triangleleft B_C^{K_{CP}}(pID, N_{RP})$ (D49 e P9 em R5d e R5a)

D52. $CP \triangleleft \{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{C-CP}}$ (CP recebe M9)

D53. $CP \equiv C \mid\sim B_C^{K_{CP}}(pID, N_{RP}), info$ (D52 e P10 em R1)

D54. $CP \triangleleft B_C^{K_{CP}}(pID, N_{RP})$ (D52 e P10 em R5d e R5a)

D55. $CP \triangleleft info$ (D52 e P10 em R5d e R5a)

Derivações da mensagem M10:

D56. $CP \mid\sim \left\{\{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP-1}}\right\}_{K_{C-CP}}$ (por M10)

D57. $CP \triangleleft \left\{\{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP-1}}\right\}_{K_{C-CP}}$ (Pelo significado de $\mid\sim$ em D56)

D58. $CP \triangleleft \{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP^{-1}}}$ (D57 e P10 em R5d)

D59. $C \triangleleft \left\{\{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP^{-1}}}\right\}_{K_{C-CP}}$ (C recebe M10)

D60. $C \equiv CP \mid\sim \{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP^{-1}}}$ (D56 e P9 em R1)

D61. $C \triangleleft \{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP^{-1}}}$ (D56 e P9 em R5d e R5a)

Aplicando-se D61, P18 e P16 em R9, temos:

D62. $\mathbf{C \triangleleft \{pID, N_{RP}, info\}_{K_{CP^{-1}}}}$ (Obtenção da assinatura desofuscada)

Derivações da mensagem M11:

D63. $C \mid\sim \left\{CP, info, \{pID, N_{RP}, info\}_{K_{CP^{-1}}}\right\}_{K_{C-IdP}}$ (por M11)

D64. $C \triangleleft \left\{CP, info, \{pID, N_{RP}, info\}_{K_{CP^{-1}}}\right\}_{K_{C-IdP}}$ (Pelo significado de $\mid\sim$ em D63)

D65. $C \triangleleft CP$ (por D64 e P5 em R5d e R5a)

D66. $IdP \triangleleft \left\{CP, info, \{pID, N_{RP}, info\}_{K_{CP^{-1}}}\right\}_{K_{C-IdP}}$ (IdP recebe M11)

D67. $IdP \triangleleft info$ (D66 e P6 em R5d e R5a)

D68. $IdP \triangleleft \{pID, N_{RP}, info\}_{K_{CP^{-1}}}$ (D66 e P6 em R5d e R5a)

D69. $IdP \equiv C \mid\sim CP, \{pID, N_{RP}, info\}_{K_{CP^{-1}}}$ (D63 e P6 em R1)

Derivações da mensagem M12:

D70. $IdP \mid\sim \left\{ CP, RP, Tk, \{Tk\}_{K_{CP^{-1}}}, \left\{RP, Tk, \{Tk\}_{K_{CP^{-1}}}\right\}_{K_{RP-IdP}}\right\}_{K_{C-IdP}}$ (por M12)

D71. $IdP \triangleleft \left\{ CP, RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}} \right\}_{K_{C-IdP}}$ ($\vdash\sim$ em D70)

D72. $IdP \triangleleft CP$ (D71 e P6 em R5d e R5a)

D73. $IdP \triangleleft Tk, \{Tk\}_{K_{CP}{}^{-1}}$ (D71 e P6 em R5d e R5a)

D74. $IdP \triangleleft \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}}$ (D71 e P6 em R5d e R5a)

D75. $C \triangleleft \left\{ CP, RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}} \right\}_{K_{C-IdP}}$ (C recebe M12)

D76. $C \triangleleft \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}}$ (D75 e P5 em R5d e R5a)

D77. $C \equiv IdP \vdash\sim CP, RP, Tk, \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}}$ (D70 e P5 em R1)

Derivações da mensagem M13 (redirecionamento):

D78. $RP \triangleleft \left\{ CP, RP, Tk, \{Tk\}_{K_{CP}{}^{-1}}, \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}} \right\}_{K_{C-RP}}$ (RP recebe M13)

D79. $RP \triangleleft CP, RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}}$ (D78 e P2 em R5d)

D80. $RP \triangleleft \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}}$ (D79 em R5a)

D81. $RP \triangleleft RP, Tk, \{Tk\}_{K_{CP}{}^{-1}}$ (D79 com R5a em P13 e R5d)

D82. $RP \triangleleft \{Tk\}_{K_{CP}{}^{-1}}$ (D81 em R5a)

D83. $RP \triangleleft Tk$ (D79 em R5a)

D84. $RP \triangleleft CP$ (D79 em R5a)

D85. $RP \equiv C \vdash\sim CP, RP, Tk, \left\{ RP, Tk, \{Tk\}_{K_{CP}{}^{-1}} \right\}_{K_{RP-IdP}}$ (D78 e P2 em R1)

Aplicando D85 em R8:

D86. $\mathbf{RP \equiv C \vdash\sim Tk}$ **(Obtenção do objetivo O3)**

Aplicando D79 e P13 em R1:

D87. $RP \models IdP \mid\sim RP, Tk, \{Tk\}_{K_{CP^{-1}}}$

D88. $RP \models IdP \mid\sim RP, Tk$ (D87 em R8)

Aplicando D88 e P17 em R3, temos que:

D89. **$RP \models IdP \models (RP,Tk)$** **(Obtenção do objetivo O2)**

Aplicando D82 e P15 em R2:

D90. $RP \models CP \mid\sim Tk$

Aplicando D90 e P17 em R3, temos que:

D91. **$RP \models CP \models Tk$** **(Obtenção do objetivo O1)**

A seguir, a Tabela 11 apresenta um resumo de todas[25] as informações obtidas (vistas $\triangleleft$) em cada entidade do sistema devido à execução do protocolo.

[25] Não é possível obter outras expressões de visualização ($\triangleleft$) de informações do usuário para as entidades do sistema por meio de derivações das mensagens do protocolo.

Tabela 11 - Informações obtidas pelas entidades do sistema devido à execução do protocolo

#	Informação	Visto por				Derivações
		C	IdP	RP	CP	
1	pID_list	◁	◁	-	-	D25, D28
2	CP_list	◁	◁	◁	-	D9, D14, D21
3	IdP	◁	◁	◁	-	D3, D5, P24
4	N_{RP}	◁	◁	◁	-	D8, D13, D20
5	pID	◁	◁	◁	-	D31, D35, D83
6	req_info	◁	◁	◁	-	D9, D14, D19
7	RP	◁	◁	◁	-	D15, D22, P23
8	$Sign_{CP}(Tk)$	◁	◁	◁	-	D62, D68, D82
9	Tk	◁	◁	◁	-	D73, D83, D13, D31,D50
10	MAC_{IdP}	◁	◁	◁	-	D74, D76, D80
11	Ts	◁	-	-	◁	D40, D46
12	Username	◁	-	-	◁	D39, D45
13	Pw	◁	-	-	◁	D41, D47
14	$B_C^{K_{CP}}(pID, N_{RP})$	◁	-	-	◁	D51, D54
15	$Sign_{CP}(B_C^{K_{CP}}(pID, N_{RP}),info)$	◁	-	-	◁	D58, D61
16	CP	◁	◁	◁	◁	D65, D72, D84, P25
17	Info	◁	◁	◁	◁	D50, D55, D67, D83

7.3 Conclusões da Análise

Com a aplicação das derivações, foi possível a obtenção das expressões enumeradas nos objetivos de autenticação do protocolo. Isso pode ser comprovado por meio das derivações D86, D89 e D91. Na análise BAN, isto implica que o protocolo é capaz de alcançar seus objetivos de autenticação de maneira bem-sucedida. No caso do protocolo NibbleID, isto significa que a verificação de atributos de identidade do usuário por parte do serviço pode ser obtida com sucesso.

Ainda, ao observar-se a Tabela 11, é possível verificar que:

- As informações listadas nas linhas de 1 a 10 são conhecidas apenas entre as entidades C, IdP e RP. Estas informações são relacionadas ao acesso do usuário a RP por meio de um pseudônimo provido por IdP .
- As informações listadas nas linhas de 11 a 15 são conhecidas apenas entre as entidades C e CP. Estas informações são relacionadas à identidade real do usuário em CP.
- As informações listadas nas linhas de 16 a 17 (info e CP) são conhecidas entre todas as entidades. Estas informações são relacionadas à comprovação de atributos do usuário. Mais especificamente, "info" representa os atributos comprovados da identidade do usuário existente em "CP" (responsável pela comprovação).

Portanto, é possível perceber que "info" e "CP" são as únicas informações em comum entre as entidades e que podem ser utilizados na tentativa de associar o acesso por meio de um pseudônimo (conhecido por RP e IdP) à identidade real do usuário (conhecida apenas por CP). Com isso, o sucesso de tal tentativa depende de fatores como o conteúdo de "info" e dos usuários em CP (esta discussão será abordada no capítulo seguinte, ao se analisar as condições de anonimato para um usuário).

Por outro lado, uma vez que o compartilhamento de "info" e "CP" deve ser aprovado e consentido pelo usuário para o acesso a um serviço, pode-se afirmar que a **privacidade** do usuário é preservada no protocolo. De fato, de acordo com a definição de privacidade apresentada na seção 3.1, em um SGI com suporte à privacidade, o usuário deve ser capaz decidir quais informações sobre sua identidade devem ser comunicadas a um serviço em particular durante uma tentativa de acesso. Em outras palavras, o protocolo não deve compartilhar qualquer informação sobre a identidade do usuário além daquelas consentidas por ele.

7.4 Observações sobre a análise BAN para o modo de atributos únicos

Esta seção discute uma possível análise BAN para o modo de **atributos únicos**, de acordo com a descrição apresentada na seção 6.3.4, quando comparada à análise para o

modo de **atributos múltiplos** por credencial. As notações utilizadas nesta seção seguem as anteriormente utilizadas na descrição do modo de **atributos múltiplos** (seção 7.2).

De modo geral, os resultados obtidos em uma possível análise deste modo não alteram as conclusões a respeito do protocolo em relação a sua capacidade de alcançar os objetivos de autenticação e o tipo de informação compartilhado entre as entidades. Por isso, optou-se por apenas discutir quais diferenças se aplicariam a esse caso. Estas diferenças são listadas a seguir.

- Na descrição dos objetivos do protocolo, nas expressões O1 e O3, Tk deve ser substituído por: $Tk_1, Tk_2, \ldots, Tk_n$;
- O campo *info* deve ser removido da análise para as derivações da mensagem M9;
- Nas derivações de M10, as expressões $\{B_C^{K_{CP}}(pID, N_{RP}), info\}_{K_{CP^{-1}}}$ devem ser substituídas por: $\{B_C^{K_{CP}}(pID, N_{RP}), att_1\}_{K_{CP^{-1}}}$, $\{B_C^{K_{CP}}(pID, N_{RP}), att_2\}_{K_{CP^{-1}}}, \ldots, \{B_C^{K_{CP}}(pID, N_{RP}), att_n\}_{K_{CP^{-1}}}$;
- Em M11, as derivações devem ocorrer a partir da alteração de: $info, \{pID, N_{RP}, info\}_{K_{CP^{-1}}}$ para: $att\_sel_1, \{pID, N_{RP}, att\_sel_1\}_{K_{CP^{-1}}}$, $att\_sel_2, \{pID, N_{RP}, att\_sel_2\}_{K_{CP^{-1}}}, \ldots, att\_sel_m, \{pID, N_{RP}, att\_sel_m\}_{K_{CP^{-1}}}$;
- Para as derivações de M12 e M13, as expressões $Tk, \{Tk\}_{K_{CP^{-1}}}$ devem ser substituídas por: $Tk_1, \{Tk_1\}_{K_{CP^{-1}}}, Tk_2, \{Tk_2\}_{K_{CP^{-1}}} \ldots, Tk_n, \{Tk_n\}_{K_{CP^{-1}}}$;
- Na Tabela 11, *info* deve ser substituído por $att_1, att_2, \ldots, att_n$;
- Do mesmo modo, Tk deve ser substituído por $Tk_1, Tk_2, \ldots, Tk_n$ na Tabela 11;
- Ainda na Tabela 11, uma linha deve ser adicionada para a informação $att\_sel_1, att\_sel_2, \ldots, att\_sel_m$. Estes valores são vistos pelas entidades C e CP somente.

7.5 Resumo do Capítulo

Este capítulo apresentou uma análise formal do protocolo por meio da lógica BAN. Por meio desta análise, foi possível verificar que os objetivos do protocolo em relação à comprovação de atributos de identidade do usuário a um serviço são cumpridos. Além disso, foi possível demonstrar que o protocolo não transmite informações não consentidas e que possam ser utilizadas por uma entidade do sistema para associar o pseudônimo à identidade real do usuário. Em outras palavras, foi possível demonstrar que a privacidade do usuário é preservada durante a execução do protocolo.

É importante observar, porém, que a preservação da **privacidade** do usuário pelo protocolo não garante que o usuário possa se manter **anônimo** em seus acessos aos serviços. Com isso, o próximo capítulo analisa e discute as condições de anonimato de um usuário no uso do sistema NibbleID, além de comparar os graus de anonimato possíveis para esse sistema e seu *overhead* criptográfico em relação a outras soluções.

8 ANÁLISE DO GRAU DE ANONIMATO NO NIBBLEID

"Real freedom is having nothing. I was freer when I didn't have a cent. Do you know what I do sometimes? Put on a ski mask and dress in old clothes, go out on the streets and beg for quarters"
Mike Tyson- ex. pugilista

Embora os conceitos de privacidade e anonimato sejam distintos entre si, conforme apresentado nas seções 3.1 e 3.2, pode-se afirmar que a utilização de mecanismos de suporte à privacidade aumenta as possibilidades do usuário estar anônimo no sistema (HEUPEL, 2010).

Dessa forma, este capítulo estuda os graus de anonimato possíveis para um usuário por meio dos mecanismos de privacidade fornecidos pelo NibbleID. Como parte desse estudo, será analisado também o *overhead* criptográfico introduzido no sistema para permitir tais graus de anonimato.

Esse estudo ajuda, também, a observar que o NibbleID tem objetivos distintos quando comparado com outras soluções de SGIs. Segundo investigação realizada no escopo deste trabalho, o NibbleID é a primeira proposta deste tipo de sistema[26] a se posicionar na lacuna entre, de um lado, soluções com pouca ou nenhuma preocupação com a privacidade do usuário e, de outro lado, soluções que provêm um alto nível de anonimato baseados em algoritmos de *Zero-Knowledge Proof* com alta demanda de processamento.

8.1 Anonimato em SGIs: preliminares

Em SGIs sem suporte à privacidade do **usuário**, as asserções de identidade enviadas por meio de um IdP a um **serviço**, de maneira geral, carregam alguma informação (um número de registro do usuário, por exemplo) que podem identificar unicamente o **usuário** dentro de um conjunto de outros usuários do sistema. Nesses casos, o sistema

[26] Sistemas de gerenciamento de identidade com comprovação de atributos e suporte à privacidade.

adiciona informações que causam um impacto negativo na privacidade e **impedem** qualquer possibilidade de anonimato de um usuário.

Por outro lado, no caso de SGIs com suporte à privacidade (ou privacidade melhorada) como o NibbleID, é esperado que o impacto na privacidade do usuário pelo simples uso do sistema seja nulo ou o menor possível. Em outras palavras, o sistema **não deve adicionar** informações que prejudiquem a privacidade.

Para efeito de ilustração, considere um baile à fantasia onde seus participantes não podem ser identificados devido às suas máscaras e vestimentas (considerando estes suficientes para esconder os atributos físicos dos convidados). Embora seja possível afirmar que o baile propicie a participação de um indivíduo de maneira anônima, continua sendo possível, por exemplo, o rastreamento deste indivíduo da festa em seu caminho de volta para casa. Essa ação possibilitaria o levantamento de informações que revelem sua identidade (por exemplo, o endereço onde mora e o carro que dirige). Assim, embora o baile propicie o anonimato para os seus participantes, pouco pode ser feito a respeito de informações inferidas por observadores na análise de fatores externos ao baile.

Com isso, é fundamental compreender que um SGI não pode **impedir** que informações sejam **inferidas** a partir de meios externos ao sistema ou mesmo a partir de informações providas pelos próprios usuários. Como exemplos, podem ser citados:

- Um usuário intencionalmente provê alguma informação ao serviço que o identifica unicamente (ex.: o número do documento de identidade);
- O conjunto de usuários do sistema não é adequado para prover o anonimato do usuário (ex.: um usuário apresenta uma credencial que o comprova como sendo do sexo masculino, porém o grupo restante é composto apenas por usuários do sexo feminino);
- Um usuário foi rastreado pelo seu endereço IP por não ter utilizado uma solução de anonimização por meio de redes *overlay*.
- O nível/taxa de utilização desse sistema pelos usuários é muito baixo (ex.: embora o sistema possua diversos usuários registrados, a requisição de credenciais é de fato feita somente por um usuário, ou seja, existe apenas um usuário ativo no sistema);

- Existe algum conhecimento privilegiado a respeito do usuário por meios externos ao sistema (ex.: ataques de engenharia social que obtenham a informação que um determinado usuário acessou o serviço);

Outro modo de se refletir sobre esse mesmo problema é observar que, da mesma forma que o nível de segurança de um sistema é dado pelo nível de segurança do seu elo mais fraco, o nível de anonimato de um sistema em rede é dado pelo menor nível de anonimato existente entre as camadas/partes do sistema. Uma vez que o NibbleID (e também os trabalhos relacionados) lida apenas com as questões de privacidade da camada de aplicação, o anonimato do usuário é dependente também dos níveis de anonimato providos pelas camadas inferiores (e.g., os mecanismos de anonimização por meio de redes *overlay*) e também das camadas superiores (e.g., os dados fornecidos pelos usuários).

8.2 Modelo de Estudo Proposto

Para a análise do anonimato no NibbleID é proposto um modelo simplificado do sistema baseado no modelo genérico para sistemas com suporte a anonimato apresentado em (DIAZ, 2006). Esse modelo é apresentado na Figura 16 a seguir.

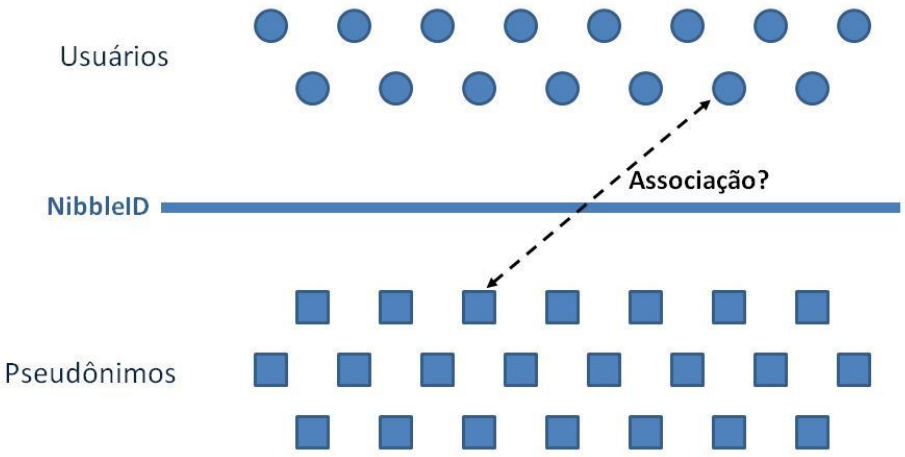

Figura 16 – Modelo utilizado para o estudo do anonimato no NibbleID

Nesse modelo, a análise do anonimato é baseada no estudo da **probabilidade** em se **associar** a **identidade real** de um usuário registrado em um determinado CP com um **pseudônimo** apresentado a um RP.

Essa probabilidade de associação é dada em função do número de elementos (cardinalidade) do **conjunto de anonimato** (*anonymity set*) na utilização de um determinado pseudônimo. O **conjunto de anonimato** é compreendido como o conjunto de possíveis usuários que podem ser diretamente associados a um dado pseudônimo (e.g,, por possuírem o mesmo conjunto de atributos associado ao pseudônimo apresentado). A Figura 17 ilustra um exemplo de um conjunto de anonimato relativo a um pseudônimo. Quanto maior for o conjunto de anonimato, maior será o nível de anonimato possível para um usuário, i.e., mais difícil será distinguir o usuário no grupo. A noção de conjunto de anonimato é fundamental em métricas de anonimato (DIAZ, 2006).

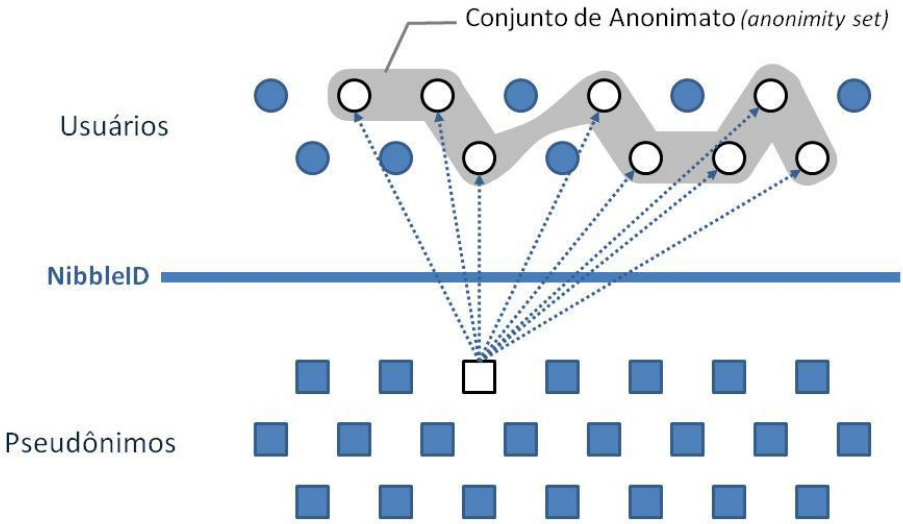

Figura 17 - Conjunto de anonimato

O presente trabalho considera dois tipos de conjuntos de anonimato distintos, de acordo com o tipo de análise efetuada: análise de atributos de identidade e análise de tempo (de relógio). As definições mais precisas para cada um desses tipos de conjuntos de anonimato serão apresentadas nas seções 8.5 e 8.6.

8.3 Métrica para medida do anonimato

Considerando o modelo apresentado na seção anterior, esta seção apresenta a métrica a ser utilizada para medir o anonimato do NibbleID. Essa métrica foi introduzida em (REITER; RUBIN, 1998) para a análise do anonimato do protocolo *Crowds*, o qual provê privacidade para mensagens nos níveis de rede e transporte por meio de uma rede *overlay*, e foi também utilizada por (MARTUCCI, 2009) para a análise de anonimato em redes *ad-hoc.* Nessa métrica, o nível de anonimato é dado em função da cardinalidade dos conjuntos de anonimato.

Com isso, define-se o grau de anonimato A_{u_i} para um usuário $u_i \in \Theta$ como $A_{u_i} = 1 - P_{u_i}$, onde Θ é o conjunto usuários do sistema e P_{u_i} é a probabilidade de u_i ser associado a um determinado pseudônimo. O grau de anonimato A_{u_i} é apresentado em uma escala contendo pontos de interesse que variam desde *comprovadamente exposto* até *privacidade absoluta*, como apresentado na Figura 18 a seguir.

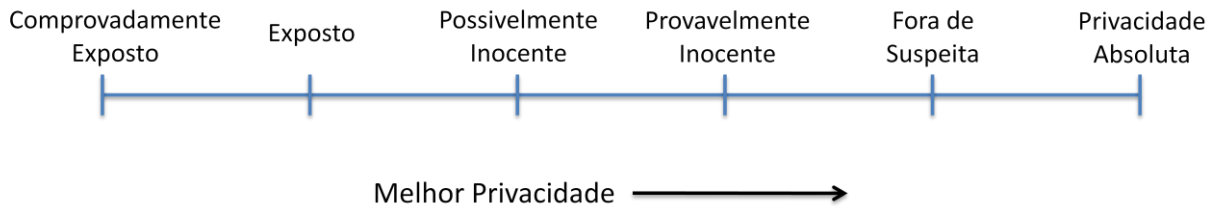

Figura 18 – Graus de anonimato, segundo (REITER; RUBIN, 1998)

Na escala apresentada na Figura 18, a definição dos pontos de interesse foram adaptados de (REITER; RUBIN, 1998) e (MARTUCCI, 2009) e, no presente trabalho, são compreendidos da seguinte maneira:

- **Privacidade Absoluta**: A probabilidade de um dado usuário $u_i \in \Theta$ ser associado a um pseudônimo em particular é zero e, portanto, $A_{u_i} = 1$;
- **Fora de suspeita**: A probabilidade de um usuário $u_i \in \Theta$ ser associado a um pseudônimo em particular é menor que a probabilidade da associação de qualquer outro usuário pertencente a Θ ser associado ao dado pseudônimo, ou seja, A_{u_i} = máximo $\{A_{u_1}, A_{u_2}, A_{u_3}, \ldots, A_{u_n}\}$;

- **Provavelmente Inocente**: A probabilidade de um dado usuário $u_i \in \Theta$ ser associado a um pseudônimo em particular é menor que $\frac{1}{2}$ e, portanto, $A_{u_i} \geq 0{,}5$;
- **Possivelmente Inocente**: Existe uma possibilidade não trivial de que um dado usuário $u_i \in \Theta$ não ser o dono de um pseudônimo em particular, de modo que $0 < A_{u_i} < 0{,}5$;
- **Exposto**: Um dado usuário $u_i \in \Theta$ pode ser associado de forma não ambígua a um pseudônimo em particular e, portanto, com $A_{u_i} = 0$;
- **Comprovadamente exposto**: Similar ao nível anterior, porém este nível adiciona o fato de ser possível uma demonstração (prova) de que o pseudônimo utilizado de fato pertence ao usuário em questão.

Da mesma forma que em (REITER; RUBIN, 1998), durante a análise deste trabalho será considerado o terceiro nível (*provavelmente inocente*) como o nível **mínimo** desejado para o grau de anonimato de um usuário. Isto garante a existência de outros usuários em condições semelhantes dentro do conjunto de anonimato.

8.4 Pontos de Análise

A análise de anonimato é feita a partir do ponto de vista das entidades do sistema, como listado a seguir:

- **RP**: Análise de A_{u_i} do ponto de vista de RP;
- **IdP**: Análise de A_{u_i} do ponto de vista de IdP;
- **CP**: Análise de A_{u_i} do ponto de vista de CP;
- **Conluio RP + IdP**: Análise de A_{u_i} do ponto de vista de RP em **conluio** com IdP com a finalidade de quebra da privacidade do usuário.
- **Conluio CP + IdP/RP**: Análise de A_{u_i} do ponto de vista de CP em conluio com IdP e/ou RP com a finalidade de quebra da privacidade do usuário.

De modo a simplificar a análise, a discussão será feita para o caso de identidades parciais geradas a partir de um **único CP**. Essa simplificação torna a análise compatível com os trabalhos relacionados a serem considerados posteriormente.

8.5 Conjunto de anonimato Γ por análise de atributos de identidade

Para uma dada **identidade parcial** associada a um **pseudônimo** e formada por uma ou mais credenciais geradas por um **mesmo CP**, define-se o **conjunto de anonimato Γ** como o **conjunto dos usuários de CP** capazes de gerar **identidades parciais** cujos **atributos de identidade** sejam exatamente iguais aos da identidade parcial associada ao pseudônimo. Em outras palavras, o conjunto de anonimato Γ representa todos os usuários de CP que podem ser considerados suspeitos pela geração de uma dada identidade parcial a partir da análise de seus atributos de identidade.

Como ilustração, considere os dois exemplos a seguir:

- Ex.1 - Uma identidade parcial criada a partir de uma credencial fornecida por um banco (CP) comprova os atributos “cliente especial” e “conta ativa” para um usuário. Nesse caso Γ é formado por todos os clientes do banco que também são “clientes especiais” e possuem “conta ativa”.
- Ex.2 - Uma identidade parcial criada a partir de uma credencial fornecida por um banco (CP) comprova o atributo “CPF: 123.456.789-00”. Nesse caso Γ é formado apenas pelo usuário que possui o CPF em questão, ou seja, o próprio dono da credencial.

8.5.1.1 Conhecimento do Conjunto de Anonimato Γ

O **conjunto de anonimato Γ** pode ser conhecido por RP e/ou IdP em diferentes níveis, os quais podem assumir valores entre o conhecimento nulo ao conhecimento total de Γ. Esse conhecimento pode ser obtido de diferentes maneiras como, por exemplo, por meio de uma relação de conluio dessas entidades com CP, de engenharia social ou de vazamento de informações. Considera-se que CP sempre possui conhecimento total de Γ, uma vez que esse conhecimento para uma dada identidade parcial pode ser inferido a

partir do conhecimento dos usuários registrados em CP e seus atributos de identidade. Com isso, para os pontos de análise em RP e/ou IdP, o estudo considerará duas situações distintas sobre o conhecimento de Γ:

- Conhecimento **total** de Γ por RP e/ou IdP;
- Conhecimento **nulo** de Γ por RP e/ou IdP.

8.6 Conjunto de anonimato Ψ por análise de tempo

Na situação particular de conluio de CP com IdP e/ou RP, a quantidade de usuários ativos com requisições no sistema pode também influenciar negativamente o grau de anonimato dos usuários. Para isso, basta a análise das informações de tempo (relógio do sistema) contidas nos *logs* das entidades de modo a filtrar e reduzir o número de elementos do conjunto de anonimato Γ. Em sistemas de anonimização do usuário por meio de redes *overlay,* essa ação é conhecida como ataques por análise de tempo (*timing attacks*) (LEVINE *et al.*, 2004) e (SHMATIKOV; WANG, M.-H., 2006). Em tais sistemas, a essência desse ataque é a busca de correlações de tempo entre as mensagens observadas em um originador e as recebidas em um destinatário (LEVINE *et al.*, 2004) .

Em SGIs com suporte à privacidade, de maneira similar, esses ataques podem ocorrer pela correlação de tempo entre as ações dos usuários onde suas identidades são conhecidas (ex.: na **geração** da credencial) com as ações onde suas identidades podem ser anônimas (ex.: na **apresentação** da credencial).

Assim, de modo a simplificar a análise nessas condições, define-se, para uma dada identidade parcial associada a um pseudônimo, **Ψ** como o **conjunto de anonimato** gerado a partir dos possíveis **usuários** encontrados em associações **por análise de tempo**. Esse conjunto pode ser obtido por meio da observação dos usuários que efetuaram requisições em CP durante a janela de tempo compreendida entre a requisição de acesso e a apresentação das credenciais em um RP/IdP. Dessa forma, o conjunto de anonimato **Ψ** será considerado como **conhecido** pelas entidades sempre que houver uma situação de conluio entre CP e IdP e/ou RP.

8.7 Análise dos graus de anonimato no NibbleID

A Tabela 12, a seguir, apresenta o resumo da análise de anonimato do modelo proposto para o NibbleID. A explicação da tabela e dos valores apresentados será feita nas subseções a seguir. Essas subseções estão organizadas de acordo com uma ordem considerada didática para a descrição de cada uma das condições (casos de "a" até "h"), ao invés da ordem utilizada na tabela.

Tabela 12 – Resumo da análise dos graus de anonimato no NibbleID

CONHECIMENTO DE Γ E PONTO DE ANÁLISE			A_{u_i}	GRAU DE ANONIMATO
Conhecimento nulo de Γ	RP	*(a)*	1	Privacidade Absoluta
	IdP	*(b)*	1	Privacidade Absoluta
	Conluio RP + IdP	*(c)*	1	Privacidade Absoluta
Conhecimento total de Γ	RP	*(d)*	$\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$	Provavelmente Inocente se $\lvert\Gamma\rvert \geq 2$
	IdP	*(e)*	$\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$	Provavelmente Inocente se $\lvert\Gamma\rvert \geq 2$
	Conluio RP + IdP	*(f)*	$\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$	Provavelmente Inocente se $\lvert\Gamma\rvert \geq 2$
	CP	*(g)*	1	Privacidade Absoluta
	Conluio CP + IdP/RP	*(h)*	$\frac{\lvert\Gamma \cap \Psi\rvert - 1}{\lvert\Gamma \cap \Psi\rvert}$	Provavelmente Inocente se $\lvert\Gamma \cap \Psi\rvert \geq 2$

Legenda

$\mathbf{A_{u_i}}$ Grau de anonimato de um usuário $\mathbf{u_i}$ (probabilidade)

Γ Conjunto de anonimato de usuário dado pelas correlações feitas por **análise dos atributos de identidade**

Ψ Conjunto de anonimato de usuário dado pelas correlações feitas por **análise de tempo**

8.7.1 Anonimato em RP e IdP - Conhecimento Total de Γ (casos d, e, f)

Do ponto de vista de RP ou IdP com o conhecimento do grupo de anonimato Γ, a probabilidade de associação P_{u_i} é dada pelo inverso de |Γ| (onde |Γ| é o número de elementos de Γ), ou seja, $P_{u_i} = \frac{1}{|\Gamma|}$. Portanto, o grau de anonimato A_{u_i} é igual a $1 - \frac{1}{|\Gamma|} = \frac{|\Gamma|-1}{|\Gamma|}$. O conluio entre RP e IdP não altera essa probabilidade, uma vez que o mesmo não adiciona nenhuma informação a respeito do usuário para as entidades.

8.7.2 Anonimato em RP e IdP - Conhecimento Nulo de Γ (casos a, b, c)

Sem nenhum conhecimento de Γ, do ponto de vista de RP ou IdP, a chance de associação P_{u_i} pode ser considerada nula, uma vez que, dentro do modelo de estudo proposto, não é possível a associação de um pseudônimo a um usuário (membro de CP pertencente a Γ). O conluio entre RP e IdP também não altera essa probabilidade, uma vez que o mesmo não adiciona nenhuma informação a respeito do usuário para as entidades. Logo, o grau de anonimato A_{u_i} do ponto de vista de RP e/ou IdP é igual a 1 para o caso de conhecimento nulo de Γ.

8.7.3 Anonimato em CP - Conhecimento Total de Γ (caso g)

Do ponto de vista de CP, embora o usuário que requisita uma credencial seja conhecido, dentro do modelo de estudo proposto, não é possível a associação desse usuário com o pseudônimo da credencial (uma vez que o pseudônimo é ofuscado para CP). Nesse modelo, o conhecimento do conjunto de possíveis pseudônimos a serem associados ao usuário só pode ser conhecido por CP por meio do conluio com RPs ou IdPs (descrito no próximo item). Logo, pode-se considerar que $P_{u_i} = 0$ nesse caso e, portanto, A_{u_i}= 1.

8.7.4 Grau de anonimato em CP em conluio com RP e/ou Idp - Conhecimento Total de Γ (caso h)

Uma vez que no NibbleID a credencial não é armazenada no dispositivo do usuário após a sua geração, a apresentação dessa credencial em um dado RP tende a ocorrer logo após a emissão da mesma por um CP. Isso resulta em uma curta janela de tempo entre as ações de geração e apresentação de credencial. Essa janela pode ser aumentada pelo usuário bastando que o mesmo demore a entrega da credencial em RP (no penúltimo passo do protocolo). Esse atraso proposital, porém, não é uma ação conveniente para o usuário Web. Com isso, considerando-se uma curta janela de tempo, o anonimato de um usuário nessas condições de conluio passa a depender fortemente de uma alta utilização da rede por outros usuários do conjunto de anonimato.

Portanto, é possível afirmar que o anonimato do NibbleID é mais vulnerável a ataques de análise de tempo nas condições onde há o conluio entre CP e RP/IdP e baixa utilização do sistema.

Com isso, o conjunto de anonimato utilizado para a análise é dado pela intersecção do conjunto Γ (relativo aos atributos de identidade parcial) e o conjunto Ψ (relativo às associações por análise de tempo). Logo, a probabilidade de associação P_{u_i} é igual a $\frac{1}{|\Gamma \cap \Psi|}$ e, portanto, o grau de anonimato é dado por: $\frac{|\Gamma \cap \Psi|-1}{|\Gamma \cap \Psi|}$.

8.8 Considerações sobre ataques de análise de tempo nas situações de conluio entre CP e IdP/RP

Observando-se a Tabela 12, é possível perceber que o requisito de maior exigência para a condição de *provavelmente inocente* é dado por $|\Gamma \cap \Psi| \geq 2$, na situação de conluio entre CP e IdP/RP (última linha da Tabela 12). A maior dificuldade na garantia desse requisito deve-se ao fato do grupo de anonimato Ψ ser dependente da utilização no sistema, a qual deve no mínimo cumprir a condição $|\Gamma \cap \Psi| \geq 2$.

Embora essa condição com o conluio de entidades apresente o menor grau de anonimato no sistema (ou o requisito mais difícil a ser cumprido), é importante ressaltar que, de acordo com (BERTHOLD; FEDERRATH; KÖHNTOPP, 2000), soluções de anonimização por redes *overlay* (tipicamente utilizadas para as camadas de enlace e rede) apresentam-se vulneráveis a ataques de análise de tempo quando os enlaces de origem e destino podem ser observados. Em outras palavras, mesmo com um mecanismo mais resistente a ataques de tempo na camada de aplicação (onde se situam as soluções de SGI), é possível que o grau de anonimato do sistema como um todo seja prejudicado pelos mecanismos de camadas inferiores nas situações de conluio total entre as entidades (enlaces podem ser observados) e baixa utilização da rede.

Ainda, embora não seja parte do objetivo desse trabalho, é interessante mencionar que, de acordo com (PEYTON; DOSHI; SEGUIN, 2007), as situações de conluio podem ser desencorajadas por meio de aplicações de políticas de privacidade e auditoria em entidades do sistema. Por exemplo, a simples aplicação de uma política para não permitir o armazenamento de *logs* em CP pode ser suficiente para evitar a existência de dados que possam ser utilizados para ataques de análise de tempo.

8.9 Análise aplicada ao ambiente de uma universidade

Para efeitos de ilustração da aplicação da análise de anonimato do NibbleID, considere-se um cenário onde o sistema é adotado pela Universidade de São Paulo (USP) e disponível para seus alunos. Considere-se, também, a existência de serviços e *sites* diversos que suportam o NibbleID e aceitam credenciais geradas pela USP. A Tabela 13, a seguir, apresenta o número de alunos matriculados nos diversos cursos na Universidade de São Paulo no primeiro semestre de 2010 (USP, 2010).

Tabela 13 - Alunos matriculados na USP em 2010

CURSOS	ALUNOS MATRICULADOS
Graduação	57.300
Mestrado	13.467
Doutorado	13.101
Especiais	5.094
Total	**88.962**

Nesse cenário, considere-se um aluno de graduação (u_i) que apresenta uma identidade parcial para um serviço contendo os atributos ***aluno regular*** e ***graduação*** a partir de um IdP de sua escolha. Nesse caso, pode-se inferir pela tabela que o tamanho do conjunto de anonimato por atributos de identidade ($|\Gamma|$) é igual a 57.300 (matriculados na graduação). Então, pelas expressões apresentadas na seção 8.3, pode-se obter o grau de anonimato do usuário para cada uma das condições exemplificadas a seguir:

- Do ponto de vista do *Site* ou serviço (com acesso à lista de alunos da USP - conhecimento de Γ) com ou sem conluio com IdP: $A_{u_i} = 0{,}99998$ (*provavelmente inocente*);
- Do ponto de vista da USP: $A_{u_i} = 1$ (*privacidade absoluta*) - não é possível saber onde a identidade foi apresentada ou o grupo de pseudônimos prováveis;
- Ponto de vista de um conluio entre a USP e o *site* ou serviço onde a identidade parcial foi apresentada: *provavelmente inocente* ($A_{u_i} \geq 0{,}5$) se **ao menos** 0,0035% dos alunos estiverem com requisições de credenciais pendentes no sistema durante o acesso do usuário em questão (aproximadamente 2 alunos).

8.10 Comparativo dos graus de anonimato no NibbleID, OpenID e U-Prove

Essa seção apresenta uma comparação dos graus de anonimato do NibbleID com outros trabalhos relacionados. Os dois trabalhos selecionados foram o OpenID e o U-Prove. Embora esses trabalhos sejam soluções distintas para finalidades distintas, essa escolha justifica-se por serem os mais próximos das características do NibbleID considerando-se quesitos de simplicidade da solução (OpenID) e provimento de privacidade (U-Prove).

8.10.1 Graus de anonimato no OpenID

No OpenID, a entidade IdP (equivalente ao OP, na nomenclatura do OpenID) é a entidade que possui a identidade do usuário e a responsável pelo provimento pseudônimos. No OpenID não existe uma entidade equivalente a CP (esse papel seria também assumido por IdP). Uma vez que o OpenID não foi projetado com base em requisitos de privacidade ou anonimato em mente, todas as requisições do usuário aos diversos *sites* ou serviços na Internet são de conhecimento de IdP. Logo, para qualquer condição onde IdP estiver presente (sozinho ou em conluio com RP), a probabilidade P_{u_i}de associação do pseudônimo com a identidade do usuário será igual a 1 e, portanto, A_{u_i}= 0.

Do ponto de vista de RP, mesmo quando não houver conluio com IdP, a associação do usuário com o pseudônimo pode ser feita caso exista o conhecimento do conjunto de anonimato Γ. Então, nesse caso, tem-se que A_{u_i}é igual a 0. Porém, se não houver nenhuma informação a respeito de Γ, pode-se considerar que a probabilidade de associação seja nula (da mesma forma como no NibbleID), uma vez que não há um conjunto de usuários os quais possam ser associados ao pseudônimo utilizado. Portanto, tem-se que para RP sem o conhecimento de Γ o valor de A_{u_i}é igual a 1.

8.10.2 Graus de anonimato no U-Prove

Uma análise completa do anonimato em sistemas que utilizam o mecanismo de *Zero Knowledge Proof* pode ser bastante complexa, uma vez que esse mecanismo possibilita testes sofisticados a respeito de atributos de identidade de um usuário. Por meio desse mecanismo, são possíveis afirmações a respeito de um usuário tais como:

- O usuário possui uma probabilidade de ~97% de ser um estudante matriculado em uma universidade;
- A probabilidade do usuário possuir o CPF igual a 123.456.789-00 é de ~99,9%.

Com isso, é possível concluir que em um sistema como o U-Prove, usuários pertencentes a um mesmo conjunto de anonimato podem possuir diferentes probabilidades de associação P_{u_i}, como apresentado na Figura 19. Assim, o cálculo de P_{u_i} deve levar em conta também as probabilidades comprovadas para cada um dos atributos de identidade e não somente o número de usuários do conjunto de anonimato (como no NibbleID). Essa característica dos mecanismos de *Zero Knowledge Proof* permite níveis mais elevados de anonimato. Por exemplo, para o caso onde o usuário possui ~90% de chance de ser um estudante, existe ainda uma dúvida (chance de ~10%) se tal usuário é de fato um estudante ou não.

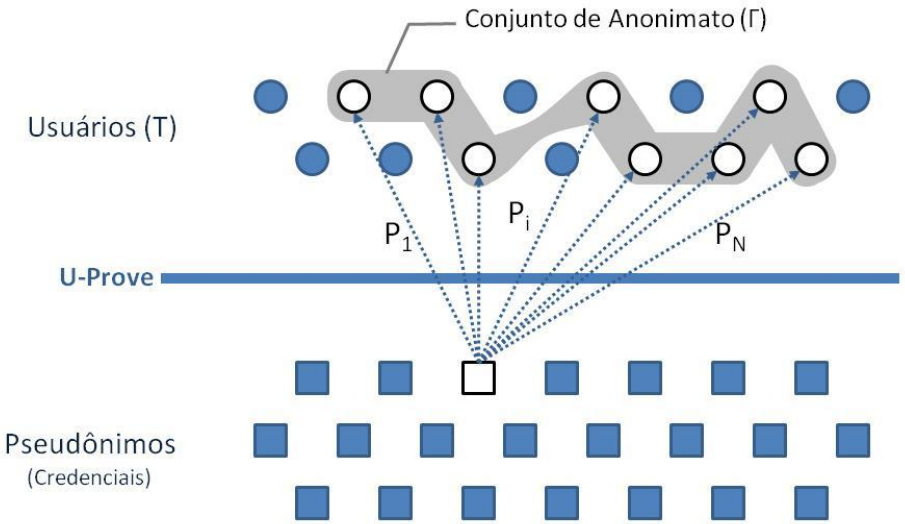

Figura 19 - Conjunto de anonimato no U-Prove com probabilidades de associação distintas para cada usuário.

Considerem-se então duas situações extremas no U-Prove. Por um lado, considere-se a situação onde todos os atributos apresentados por um usuário são comprovados com probabilidade igual a 100%. Nesse caso, o grau de anonimato do U-Prove tende a ser minimizado e a análise pode ocorrer da mesma forma que no NibbleID, utilizando-se o tamanho dos conjuntos de anonimatos. No lado oposto, considere-se a situação onde todos os atributos são apresentados com probabilidade igual a 0%. Nesse caso, o grau de anonimato do U-Prove tende a ser maximizado (nenhum atributo foi revelado) e o grupo de anonimato Γ torna-se igual ao conjunto total de usuários de CP.

Com isso, é possível afirmar que o grau de anonimato A_{u_i} no U-Prove, do ponto de vista de RP, pode variar entre $\frac{|\Gamma|-1}{|\Gamma|}$ e $\frac{|\mathbf{T}|-1}{|\mathbf{T}|}$, onde que **T** é o conjunto total de usuários de CP. Aplicando-se a mesma análise, em situações de conluio onde a análise de tempo torna-se possível, tem-se que A_{u_i} pode variar entre $\frac{|\Gamma\cap\Psi|-1}{|\Gamma\cap\Psi|}$ e $\frac{|\mathbf{T}\cap\Psi|-1}{|\mathbf{T}\cap\Psi|}$. Embora essa expressão seja similar à aplicada no NibbleID nessa mesma situação, o conjunto Ψ tende a ser maior no caso do U-Prove, uma vez que as credenciais nesse sistema podem ser armazenadas no usuário para uso posterior.

8.10.3 Comparativo das Soluções

A Tabela 14, a seguir, apresenta um resumo comparativo para os valores de A_{u_i} em função dos conjuntos de anonimato para o OpenID, NibbleID e U-Prove.

Pode-se notar que algumas das condições analisadas no NibbleID não se aplicam para o OpenID e U-Prove. Isso ocorre, pois o OpenID não possui a entidade CP em sua arquitetura (as funções dessa entidade são atribuídas ao provedor de identidade - IdP). Já na nomenclatura do U-Prove não existe a entidade IdP, a entidade RP é conhecida como *Verifier* (verificador) e a entidade CP como *Issuer* (emissor).

Tabela 14 - Comparação ilustrativa dos graus de anonimato do OpenID, NibbleID e U-Prove

CONHECIMENTO DE Γ E PONTO DE ANÁLISE		VALOR DE A_{u_i}		
		OpenID	NibbleID	U-Prove
Conhecimento nulo de Γ	RP	1	1	1
	IdP	*	1	***
	Conluio RP + IdP	*	1	***
Conhecimento total de Γ	RP	0	$\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$	De $\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$ a $\frac{\lvert T\rvert - 1}{\lvert T\rvert}$
	IdP	0	$\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$	***
	Conluio RP + IdP	0	$\frac{\lvert\Gamma\rvert - 1}{\lvert\Gamma\rvert}$	***
	CP	**	1	1
	Conluio CP + IdP/RP	**	$\frac{\lvert\Gamma \cap \Psi^{\text{NibbleID}}\rvert - 1}{\lvert\Gamma \cap \Psi^{\text{NibbleID}}\rvert}$	De $\frac{\lvert\Gamma\cap\Psi^{\text{UProve}}\rvert-1}{\lvert\Gamma\cap\Psi^{\text{UProve}}\rvert}$ a $\frac{\lvert T\cap\Psi^{\text{UProve}}\rvert-1}{\lvert T\cap\Psi^{\text{UProve}}\rvert}$

* Não se aplica pois no OpenID a entidade IdP possui também as funções de CP e, portanto, possui acesso às informações relativas à identidade dos usuários. Logo, na situação onde IdP estiver presente, **não existe** a condição de **conhecimento nulo de Γ**.

** Não se aplica, pois a entidade CP é inexistente em OpenID.

*** Não se aplica, pois a entidade IdP é inexistente em U-Prove.

Legenda

T — Conjunto total de usuários do sistema (que podem gerar credencial)

Γ — Conjunto de anonimato dado pelos atributos de identidade. $\Gamma \subset T$

Ψ^{NibbleID} — Conjunto de anonimato dado pelas correlações feitas por análise de tempo no NibbleID. $\Psi^{\text{NibbleID}} \subset T$

Ψ^{UProve} — Conjunto de anonimato dado pelas correlações feitas por análise de tempo no U-Prove. O conjunto Ψ^{UProve} tende a ser maior que Ψ^{NibbleID} uma vez que as credenciais U-Prove são armazenadas no dispositivo do cliente para uso posterior. $\Psi^{\text{UProve}} \subset T$

Observando a Tabela 14 é possível notar que o grau de anonimato e, portanto, a privacidade no NibbleID é sempre maior que no OpenID. Por outro lado, o NibbleID, de modo geral, não alcança os mesmos níveis de anonimato e privacidade possíveis no U-Prove. Essa conclusão reflete o nível de complexidade, especialmente criptográfico, das soluções analisadas. A seção a seguir provê uma análise do *overhead* criptográfico para o NibbleID e o U-Prove.

8.11 Overhead criptográfico

Esta seção compara o *overhead* criptográfico do NibbleID com o U-Prove por meio da análise do número de operações matemáticas necessárias para a execução dos protocolos no lado do cliente para cada uma das soluções.

A análise do *overhead* gerado por essas operações criptográficas é importante, pois, em alguns casos, o *overhead* criptográfico pode impossibilitar o uso de uma determinada solução em alguns cenários. Por exemplo, o Idemix pode ser inapropriado para dispositivos de baixa capacidade de processamento. De acordo com os criadores do protocolo em (CAMENISCH, JAN; HERREWEGHEN, VAN, 2002), a medida de tempo de processamento para as operações autenticação e apresentação de uma credencial podem chegar a um total de 30 segundos[27] em um Pentium III 1.1 GHz. Em um dispositivo móvel de baixa capacidade computacional, esse tempo tende a ser ainda mais elevado e não adequado para aplicações Web na Internet.

Em um SGI, dado que um usuário pode acessar o sistema por meio de um dispositivo móvel de baixa capacidade de processamento, o cliente é potencialmente a entidade na arquitetura do sistema mais sensível ao *overhead* gerado pelas operações criptográficas do protocolo. Por esse motivo, a análise dessa seção concentra-se nessa entidade do sistema a qual também é comum às duas soluções analisadas (NibbleID e U-Prove).

[27] Detalhes da implementação e medição de tempos podem ser encontrados em (CAMENISCH, JAN; HERREWEGHEN, VAN, 2002).

A escolha do U-Prove nessa comparação (ao invés do Idemix) ocorreu-se devido ao U-Prove demandar menos recursos de processamento que o Idemix[28] (MOSTOWSKI; VULLERS, 2011). Com isso, uma solução comprovadamente mais leve que o U-Prove (no caso, o NibbleID) será também mais leve que o Idemix.

Para a análise das operações criptográficas no NibbleID, considerou-se o esquema de assinaturas parcialmente cegas proposto por (ZHANG; SAFAVI-NAINI; SUSILO, 2003b), onde são utilizadas técnicas de emparelhamento bilinear para a otimização na geração das assinaturas. Optou-se pelo modo de atributos únicos por credencial do NibbleID por ser o modo com maior custo de processamento (pior caso para o NibbleID). Já para a análise do U-Prove, utilizou-se a especificação descrita em (PAQUIN, 2011), disponibilizada pela Microsoft, que utiliza técnicas de curvas elípticas e Z*ero Knoledge Proof*. Para a contagem do número de operações no U-Prove optou-se, quando necessário, pelas condições mais otimistas (com o menor número de operações). Com isso, a comparação entre o NibbleID e o U-Prove tende a mostrar a condição com menor discrepância existente entre os dois sistemas (melhor caso para o U-Prove).

A Tabela 15, a seguir, apresenta o **número de operações** no cliente para as soluções NibbleID e U-Prove nos seguintes procedimentos: geração de credencial e primeiro uso em um serviço e usos subsequentes em um mesmo serviço (sem a reapresentação de credencial). Na maioria dos casos na tabela, os números de operações matemáticas de cada solução são expressos por meio de duas variáveis: $\mathbf{N_{atrib}}$, que representa o número de atributos apresentados pelo usuário a um RP e $\mathbf{N_{Total}}$, que representa o número total de atributos **possíveis** de serem apresentados pelo usuário a um RP. A definição de $\mathbf{N_{Total}}$ é necessária, uma vez que no U-Prove todos os atributos ($\mathbf{N_{Total}}$) são armazenados dentro da credencial.

No NibbleID, o procedimento de "uso subsequente em um mesmo serviço" não possui *overhead* criptográfico, uma vez que a operação de comprovação de pseudônimo é executada pelo OpenID e a reapresentação de credenciais pode ser dispensada, conforme descrito na seção 6.3.5. A tabela mostra também a **diferença** entre o número

[28] Esse fato é esperado, uma vez que o U-Prove provê menos funcionalidades que o Idemix (CAMENISCH, J.; KOHLWEISS; SORIENTE, 2010)

de operações nas duas soluções. Uma vez que o U-Prove utiliza operações criptográficas mais complexas é natural que o número de operações necessárias para sua execução seja maior.

Tabela 15 - Número de operações criptográficas no cliente para as soluções NibbleID e U-Prove

	NÚMERO DE OPERAÇÕES NO CLIENTE		DIFERENÇA NO NÚMERO DE OPERAÇÕES
	NibbleID	U-Prove	U-Prove – NibbleID
Geração da credencial e Primeiro uso em um serviço	kgen = 0 exp = 0 mul = 1 x N_{atrib} inv = 1 x N_{atrib}	kgen = 1 exp = N_{Total} + N_{atrib} + 10 mul = 2 x N_{Total} +7 inv = 3	kgen= 1 exp = N_{Total} + N_{atrib} + 10 mul = 2 x N_{Total} – N_{atrib} + 7 inv = 3 – N_{atrib}
Usos subsequentes (sem reapresentação de credencial)	exp = 0 mul = 0 inv = 0	exp = 1 x (N_{Totral} - N_{atrib}) + 1 mul = 2 x (N_{Totral} - N_{atrib}) +1 inv = 1	exp = 1 x (N_{Total} - N_{atrib}) + 1 mul = 2 x (N_{Total} - N_{atrib}) +1 inv = 1

Legenda

kgen	Operação de geração de números primos grandes utilizado pelo U-Prove para a geração de chaves por credencial
exp	Operação de exponencial
mul	Operação de multiplicação
inv	Operação de inversão
N_{atrib}	Número de atributos apresentados pelo usuário a um RP
N_{Total}	Número total de atributos possíveis de serem apresentados pelo usuário a um RP ($\mathbf{N_{Total} \geq N_{atrib}}$). No U-Prove todos os atributos são armazenados dentro da credencial.

8.12 Resumo do Capítulo

Esse capítulo apresentou uma análise dos graus de anonimato do NibbleID. Por meio dessa análise, foi possível verificar situações onde uma possível condição de anonimato do usuário torna-se favorável ou desfavorável. Foi apresentado também um estudo comparativo do anonimato e *overhead* criptográfico do NibbleID e outras soluções relevantes, no caso, o OpenID e o U-Prove.

As análises e discussões desse capítulo corroboram a idéia de que o NibbleID situa-se entre dois opostos das soluções de SGI com comprovação de atributos e suporte à privacidade, onde, de um lado, encontram-se soluções com pouca ou nenhuma preocupação com a privacidade do usuário e, de outro lado, soluções que fornecem um alto nível de anonimato baseadas em complexos algoritmos de criptografia com superior requisito de processamento.

Esse posicionamento justifica-se quando o ambiente Web é considerado, pois o uso de navegadores em dispositivos cuja capacidade de processamento pode ser limitada sugere a necessidade de mecanismos de criptografia mais simples. Já do ponto de vista do anonimato, é importante destacar que a capacidade de um sistema em prover anonimato é dada pela camada (parte do sistema), cujo grau de anonimato é menor. Assim, altos níveis de anonimato em um SGI (tipicamente na camada de aplicação) tornam-se exagerados quando estes ultrapassam os níveis máximos fornecidos por outras soluções de camadas inferiores (como no caso de anonimização por redes *overlay*) do modelo TCP/IP.

O próximo e último capítulo desta tese resume as principais características deste trabalho e suas contribuições mais relevantes, além de mostrar algumas possibilidades de pesquisas futuras não exploradas dentro do escopo desta tese.

9 CONSIDERAÇÕES FINAIS

Este capítulo apresenta as considerações finais acerca deste trabalho para a proposta de uma solução de **gerenciamento de identidades com suporte à privacidade e comprovação de atributos do usuário no Ambiente Web**. Assim, este capítulo inicia com a reapresentação dos requisitos de sistema e a descrição de como estes são atendidos na solução proposta. Em seguida, são expostos os principais resultados obtidos nesta tese, salientando-se suas contribuições e inovações. Com isso, na seção de trabalhos futuros, são apresentadas as possibilidades e direções para continuidade deste trabalho. Por fim, são listadas as publicações ou artigos previstos do autor que estão relacionados diretamente ou indiretamente com o desenvolvimento desta tese.

9.1 Atendimento aos Requisitos

A Tabela 16, a Tabela 17 e a Tabela 18, a seguir, reapresentam os requisitos especificados para o ambiente, privacidade e segurança, respectivamente. Para cada um destes requisitos é apresentada a característica do sistema NibbleID que permite o seu atendimento.

Tabela 16 - Atendimento dos requisitos de ambiente

Requisito de Ambiente	Atendimento do requisito no NibbleID
I. Escolha do IdP independente de organização	O NibbleID prevê a existência de diversos IdPs co-existindo no sistema. A escolha do IdP independe das entidades CPs as quais o usuário faz parte.
II. Possibilidade de acessos subseqüentes a partir de equipamentos diferentes	O protocolo não requer o armazenamento de informações ou *tokens* de acesso no dispositivo do cliente.
III. Uso de navegadores Web e tecnologias relacionadas no cliente	As transferências de dados entre as entidades no protocolo são efetuadas por redirecionamentos HTTP. A geração da credencial pode ser feita por *scripts, javascript, plugins* ou outras tecnologias Web relacionadas.
IV. Possibilidade de reutilização de um pseudônimo em diferentes sessões	O protocolo OpenID prevê a reutilização do pseudônimo. Essa operação no NibbleID possibilita ainda a escolha do tipo de pseudônimo a ser utilizado.
V. Compatibilidade com protocolos existentes	Utilização dos protocolos OpenID e HTTP(S).
VI. Descentralização	A arquitetura do NibbleID prevê a existência de diversas entidades de IdP e CP distintas co-existindo na Internet.
VII. Uso de recursos computacionais mínimos no cliente	Menor número de operações criptográficas quando comparado com soluções similares.

Tabela 17 - Atendimento dos requisitos de privacidade

Requisito de Privacidade	Atendimento do requisito no NibbleID
I. Não-relacionabilidade do usuário	Provido pelo mecanismo de assinatura parcialmente cega.
II. Não-relacionabilidade entre acessos	O NibbleID permite que o usuário decida se os seus acessos poderão ser relacionados ou não por meio da escolha do tipo de pseudônimo.
III. Comprovação de atributos selecionados pelo usuário	O protocolo permite que o usuário escolha e aprove quais informações devem ser compartilhados com um determinado serviço.
IV. Minimização de dados	Provido pelo mecanismo de assinatura parcialmente cega e escolha de atributos pelo usuário.

Tabela 18 - Atendimento dos requisitos de segurança

Requisito de Segurança	Atendimento do requisito no NibbleID
I. Confidencialidade	Utilização do HTTPS em todas as etapas do protocolo.
II. Prevenção a ataques de repetição	Credenciais e asserções OpenID podem ser apresentadas apenas uma vez devido ao valor de nonce gerado por RP.
III. Não-transitividade	Asserções OpenID são geradas contendo a identificação do serviço requisitado pelo usuário.
IV. Prevenção à personificação de pseudônimos	Assim como no OpenID, o IdP é o responsável por prevenir a personificação de um pseudônimo. Isso é feito por meio da associação de pseudônimos às contas de usuário em IdP.

9.2 Contribuições e Inovações

De acordo com o apresentado no capítulo introdutório desta tese, este trabalho se propôs a investigar e responder à seguinte questão de pesquisa:

*Como prover um sistema de **gerenciamento de identidades** que proteja a **privacidade** do usuário e, ao mesmo tempo, possibilite a **comprovação de atributos de identidade** do usuário para um provedor de serviços na **Web atual**?*

De modo a responder essa questão, este trabalho especificou os requisitos de ambiente, privacidade e segurança necessários para uma solução de SGI com suporte à privacidade no **ambiente particular** da Web na Internet. **A especificação de requisitos em tal ambiente é a primeira contribuição deste trabalho**.

Uma vez que no estudo de soluções no estado da arte realizado não foram encontradas soluções que atendessem aos requisitos de ambiente especificados em sua completude, este trabalho propôs o protocolo e a arquitetura NibbleID. Na seção anterior, foram apresentadas de modo resumido as características do sistema NibbleID que permitem o cumprimento de todos os requisitos de ambiente, privacidade e segurança especificados de acordo com as necessidades da proposta. Dessa forma, este trabalho **contribui e inova ao propor uma solução de um SGI com suporte à privacidade e comprovação de atributos de identidades adequada às necessidades da Web atual**.

Além do cumprimento dos requisitos do ambiente, o NibbleID **contribui ao propor uma arquitetura com papéis distintos para as entidades CP e IdP na Internet**. Essa arquitetura adequa-se às possíveis relações de confiança existentes entre usuários e provedores (sejam eles provedores de identidade, de credenciais ou de serviços) na Internet e possibilita uma maior flexibilidade na aplicação da solução. Ainda, a arquitetura possibilita que um usuário gerencie identidades parciais compostas por atributos provenientes de diferentes organizações (CPs) em um provedor de identidades (IdP) centralizado de sua escolha.

Além disso, o protocolo NibbleID **inova devido ao seu posicionamento entre dois opostos das soluções de gerenciamento de identidade com comprovação de atributos e suporte à privacidade**, onde, de um lado, encontram-se soluções com pouca ou nenhuma preocupação com a privacidade do usuário e, de outro lado, soluções que provêm um alto nível de anonimato baseados em complexos algoritmos de criptografia com superior requisito de processamento. Esse posicionamento é justificado quando consideradas as características do ambiente Web (discutido na seção 8.12).

Além das contribuições no que se refere à solução proposta do sistema NibbleID, podem-se considerar também contribuições devido aos métodos utilizados na análise do sistema. Em particular, a utilização da lógica BAN neste trabalho **inova ao ser aplicada à demonstração formal da obtenção dos objetivos de preservação de privacidade** do protocolo, no que se refere à exposição de informações por parte das entidades. Em geral, a utilização da lógica BAN na literatura restringe-se a verificação dos objetivos de autenticação de um protocolo. Ainda, a **aplicação de uma métrica de anonimato em sistemas de gerenciamento de identidades** pode também ser considerada como uma contribuição na área, uma vez que nenhuma análise similar foi encontrada na literatura para tais sistemas.

Por fim, pode-se afirmar que este trabalho **contribui para os projetos futuros** no campo de **gerenciamento de identidades com suporte à privacidade** e **autorização em serviços por meio da comprovação de atributos de identidade** do usuário no **ambiente Web**.

9.3 Trabalhos futuros

Como um trabalho futuro, pode-se considerar o estudo de mecanismos que forneçam ao usuário informações relativas ao nível de exposição de sua identidade no uso de serviços. Com tais informações, um usuário poderia decidir por apresentar, ou não, uma identidade parcial a um serviço baseado nas chances de sua identidade real ser exposta.

Ainda, com a adoção de sistemas de computação em nuvem na Internet, pode-se considerar o estudo das implicações no uso de tais sistemas na solução de SGI proposta neste trabalho. Por exemplo, mudanças no protocolo poderiam ser consideradas supondo-se a facilidade do armazenamento seguro dos dados dos usuários em servidores na nuvem (e.g.: credenciais para acessos subsequentes aos serviços).

Por fim, pode-se considerar como um trabalho futuro a adoção ou criação de um mecanismo flexível para negociação dos atributos a serem apresentados pelo usuário ao serviço. O uso de uma linguagem para essa negociação pode ser considerada, como proposto em (CAMENISCH, J. *et al.*, 2010).

9.4 Produções Relacionadas

Esta seção lista, em ordem de importância, as publicações existentes ou artigos previstos do autor que estão relacionados diretamente ou indiretamente com o desenvolvimento desta tese.

- Patente submetida: Method for privacy preserving authorization in Pervasive Environments (*Patent number: P31750*).
- Artigo em conferência: Threat Modeling an Identity Management System for Mobile Internet (DOMINICINI *et al.*, 2010).
- Artigo em Journal: "*A Lightweight Protocol for Privacy-Preserving Authentication in Pervasive Environments*" (SIMPLICIO *et al*.). Em International Journal of Communication Systems. Aceitação condicionada a nova revisão baseada nos comentários dos avaliadores (em setembro de 2011).

- Artigo em Journal: *"Cryptanalysis of an efficient three-party password-based key exchange scheme"* (SIMPLICIO, M; SAKURAGUI, R). Aceito em 27 de setembro de 2011 em International Journal of Communication Systems.
- Artigo em desenvolvimento: *"A Privacy Preserving Identity Management and Authorization System based on OpenID"* (SAKURAGUI *et al*.). Em desenvolvimento.

REFERÊNCIAS

ABDELMAJID, N. T.; HOSSAIN, M. A.; SHEPHERD, S.; MAHMOUD, K. **Improved Kerberos Security Protocol Evaluation using Modified BAN Logic**. 2010 IEEE 10th International Conference on Computer and Information Technology (CIT). **Anais**... Bradford, UK: IEEE. , 29 jul 2010.

ALLIANCE, L. **Introduction to the liberty alliance identity architecture**. Disponível em: <http://www. projectliberty. org>. Acesso em: 1 mar. 2011. 2003.

AULETE, F. J. .; VALENTE, A. L. D. . **Dicionário contemporâneo da língua portuguesa Aulete**. Lexikon Editora Digital. , 2006.

BARNES, S. B. **A privacy paradox: Social networking in the United States**. First Monday, v. 11, n. 9, p. 11–15, 2006.

BATY, U.; ATYCY, B. **Web 2.0 ou identidade 2.0: o papel das ferramentas Web 2.0 na construção da identidade dos jovens turcos**. Revista de Informática Aplicada/Journal of Applied Computing, v. 6, n. 1, 2010.

BAUER, M.; MEINTS, M.; HANSEN, M. **FIDIS Deliverable D3. 1**–Structured Overview on Prototypes and Concepts of Identity Management Systems. Frankfurt aM, 2005.

BERGHEL, H. **Hijacking the web**. Communications of the ACM, v. 45, n. 4, p. 23–27, 2002.

BERTHOLD, O.; FEDERRATH, H.; KÖHNTOPP, M. **Project "anonymity and unobservability in the Internet"**. Proceedings of the tenth conference on Computers, freedom and privacy challenging the assumptions - CFP '00. **Anais**... Toronto, Ontario, Canada. Disponível em: <http://dl.acm.org/citation.cfm?id=332211>. Acesso em: 1 set. 2011. , 2000

BHARGAV-SPANTZEL, A.; CAMENISCH, J.; GROSS, T.; SOMMER, D. **User centricity: a taxonomy and open issues**. Journal of Computer Security, v. 15, n. 5, p. 493–527, 2007.

BIANCHI, G.; BONOLA, M.; FALLETTA, V.; PROTO, F.; TEOFILI, S. **The SPARTA pseudonym and authorization system**. Science of Computer Programming, 30 set 2008.

BICHSEL, P.; CAMENISCH, J. **Mixing Identities with Ease**. Policies and Research in Identity Management, p. 1–17, 2010.

BOYAN, J. **The Anonymizer - Protecting User Privacy on the Web**. Citeseer. , 1997

BRANDS, S. **A technical overview of digital credentials**. Citeseer. , 2002

BRANDS, S. A. **Rethinking public key infrastructures and digital certificates: building in privacy**. Montreal, Canadá: The MIT Press, 2000.

BURR, W. E.; DODSON, D. F.; POLK, W. T. **Electronic authentication guideline**. NIST Special Publication, v. 800, p. 63, 2004.

BURROWS, M.; ABADI, M.; NEEDHAM, R. M. **A Logic of Authentication**. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, v. 426, n. 1871, p. 233 -271, 8 dez 1989.

CADENHEAD, R.; SMITH, G.; HANNA, J.; KEARNEY, B. **The application/rss+ xml media type**. Network Working Group (www. rssboard. org/rss-mime-type-application. txt), 2006.

CAMENISCH, J.; GROSS, T. **Efficient attributes for anonymous credentials**. Proceedings of the 15th ACM conference on Computer and communications security. **Anais**... , 2008

CAMENISCH, J.; KOHLWEISS, M.; SORIENTE, C. **Solving Revocation with Efficient Update of Anonymous Credentials**. Security and Cryptography for Networks: 7th International Conference, SCN 2010, September 13-15, 2010, Proceedings. **Anais**... Amalfi, Italy, 2010

CAMENISCH, J.; LYSYANSKAYA, A. **An efficient system for non-transferable anonymous credentials with optional anonymity revocation**. Advances in Cryptology—EUROCRYPT 2001, p. 93–118, 2001.

CAMENISCH, J.; LYSYANSKAYA, A. **A signature scheme with efficient protocols**. Security in communication networks, p. 268–289, 2003.

CAMENISCH, J.; MÖDERSHEIM, S.; NEVEN, G.; PREISS, F. S.; SOMMER, D. **A language enabling privacy-preserving access contro**l. 2010.

CAMENISCH, J; HERREWEGHEN, E. VAN. **Design and implementation of the *idemix* anonymous credential system**. Proceedings of the 9th ACM conference on Computer and communications security - CCS '02. **Anais**... Washington, DC, USA. Disponível em: <http://dl.acm.org/citation.cfm?id=586114>. Acesso em: 9 set. 2011. , 2002

CAMERON, K. **The laws of identity**. Disponível em: <http://www.identityblog.com>. Acesso em: 1 mar. 2011.

CANARD, S.; MALVILLE, E.; TRAORÉ, J. **Identity federation and privacy: one step beyond**. Proceedings of the 4th ACM workshop on Digital identity management. **Anais**..., DIM '08. New York, NY, USA: ACM. , 2008

CHAUM, D. **Blind signatures for untraceable payments**. Advances in Cryptology: Proceedings of Crypto. **Anais**..., 1983

CHOW, S.; HUI, L.; YIU, S.; CHOW, K. T**wo Improved Partially Blind Signature Schemes from Bilinear Pairings.** In: BOYD, C.; GONZÁLEZ NIETO, J. (Eds.). Information Security and Privacy. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2005. v. 3574p. 355-411.

CORELLA, F. **Pros and Cons of U-Prove for NSTIC | Pomcor**. Disponível em: <http://pomcor.com/2011/10/04/pros-and-cons-of-u-prove-for-nstic/>. Acesso em: 11 out. 2011.

DEY, A.; WEIS, S. **PseudoID: Enhancing Privacy for Federated Login**. Hot Topics in Privacy Enhancing Technologies, December, 2010.

DHAMIJA, R.; DUSSEAULT, L. **The Seven Flaws of Identity Management**: Usability and Security Challenges. Security Privacy, IEEE, v. 6, n. 2, p. 24 -29, abr 2008.

DIAZ, C. **Anonymity Metrics Revisited**. (S. Dolev, R. Ostrovsky, & Andreas Pfitzmann, Eds.)Anonymous Communication and its Applications. **Anais**..., Dagstuhl Seminar Proceedings. Dagstuhl, Germany: Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany. Disponível em: <http://drops.dagstuhl.de/opus/volltexte/2006/483>. , 2006

DIFFIE, W.; HELLMAN, M. **New directions in cryptography**. Information Theory, IEEE Transactions on, v. 22, n. 6, p. 644-654, 1976.

DINGLEDINE, R.; MATHEWSON, N.; SYVERSON, P. **Tor: The second-generation onion router**. Proceedings of the 13th conference on USENIX Security Symposium-Volume 13. **Anais**... , 2004

DOMINICINI, C.; SIMPLÍCIO, M.; SAKURAGUI, R.; CARVALHO, T. **Threat Modeling an Identity Management System for Mobile Internet**. . Rio de Janeiro, Brasil. , 2010

FEIGE, U.; FIAT, A.; SHAMIR, **A. Zero-knowledge proofs of identity**. Journal of

Cryptology, v. 1, n. 2, p. 77-94, jun 1988.

FENTON, J. **Nymwars: A possibly unpopular opinion**. Disponível em: <http://altmode.wordpress.com/2011/08/09/nymwars-a-possibly-unpopular-opinion/>. Acesso em: 29 set. 2011.

FERNANDES, R. **Google+ Terminates Fake Accounts**. Disponível em: <http://tech2.in.com/news/social-networking/google-terminates-fake-accounts/232172>. Acesso em: 29 set. 2011.

FIELDING, R.; GETTYS, J.; MOGUL, J.; FRYSTYK, H. *et al.* **RFC2616**: Hypertext Transfer Protocol–HTTP/1.1. RFC Editor United States, 1999.

GARRETT, J. J.; OTHERS. **Ajax: A new approach to web applications**. Disponível em: <http://www.adaptivepath.com/ideas/e000385>. Acesso em: 1 mar. 2011.

HAMMER-LAHAV, E.; RECORDON, D. **The oauth 1.0 protocol.** Internet Engineering Task Force (IETF) RFC5849, p. 2070–1721, 2010.

HARDT, D. **Identity 2.0**. Disponível em: <http://www.identity20.com/media/OSCON2005/>. Acesso em: 1 mar. 2011.

HARDT, D.; BUFU, J.; HOYT, J. **OpenID Attribute Exchange 1.0-Final**. 2007.

HEUPEL, M. **Porting and evaluating the performance of IDEMIX and TOR anonymity on modern smart-phones**..: http://www.uni-siegen.de/fb5/itsec/mitarbeiter/bourimi/da_heupel.pdf. , 2010

HORSTER, P.; MICHELS, M.; PETERSEN, H. **Blind multisignature schemes and their relevance to electronic voting**. Proc. 11th Annual Computer Security Applications Conference. **Anais**... , 1995

IVES, B.; WALSH, K. R.; SCHNEIDER, H. The domino effect of password reuse. **Communications of the ACM**, v. 47, p. 75–78, abr 2004.

JOSANG, A.; POPE, S. **User centric identity management**. AusCERT Asia Pacific Information Technology Security Conference. **Anais**... , 2005

KARAGODIN, A. M. **Public key infrastructure-enabled services**. The 9th Russian-Korean International Symposium on Science and Technology, 2005. KORUS 2005. Proceedings. **Anais**... IEEE. , 26 jul 2005

KISSEL, B. **OpenID 2009 Year in Review**. Disponível em: <http://openid.net/2009/12/16/openid-2009-year-in-review/>.

LABALME, F.; LINDELSEE, M.; WACHOB, G. **An Introduction to XRIs**. Citeseer. , 2005

LEBLANC, J. **Programming Social Applications: Building Viral Experiences with OpenSocial, OAuth, OpenID, and Distributed Web Frameworks**. O'Reilly Media, Inc., 2011.

LEVINE, B. N.; REITER, M. K.; WANG, C.; WRIGHT, M. **Timing Attacks in Low-Latency Mix Systems**. In: JUELS, A. (Ed.). Financial Cryptography. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. v. 3110p. 251-265.

MADDEN, M.; FOX, S. **Riding the waves of "Web 2.0"**. Backgrounder, Pew Internet and American Life Project, v. 23, n. 1, 2007.

MALIKI, T. E.; SEIGNEUR, J.-M. **A Survey of User-centric Identity Management Technologies**. Emerging Security Information, Systems, and Technologies, The International Conference on, v. 0, p. 12-17, 2007.

MALINEN, J. **Windows CardSpace**. Seminar on Network Security, Helsiki University of Technology, autumn. **Anais**.... , 2006

MARTUCCI, L. A. **Identity and Anonymity in Ad Hoc Networks**. Karlstad University, 2009.

MCCRACKEN, H. **Google+'s Real-Name Policy: Identity vs. Anonymity.** Time, undefined 2011.

MENEZES, A. J.; OORSCHOT, P. C. VAN; VANSTONE, S. A. **Handbook of applied cryptography**. CRC, 2001.

MERRELS, J. **Sxip Identity. DIX: Digital Identity Exchange Protocol**. . Internet Draft. , 2006

MIYATA, T.; KOGA, Y.; MADSEN, P.; ADACHI, S.-I. *et al.* **A Survey on Identity Management Protocols and Standards**. IEICE - Trans. Inf. Syst., v. E89-D, p. 112–123, jan 2006.

MORGAN, R. L.; CANTOR, S.; CARMODY, S.; HOEHN, W.; KLINGENSTEIN, K. Federated Security: **The Shibboleth Approach.** Educause Quarterly, v. 27, n. 4, p. 6, 2004.

MOSTOWSKI, W.; VULLERS, P. **Efficient U-Prove Implementation for Anonymous Credentials on Smart Cards**. (G. Kesidis & H. Wang, Eds.)7th International ICST Conference on Security and Privacy in Communication Networks, SecureComm 2011, London, UK, September 7-9, 2011. Proceedings. **Anais**..., Lecture Notes of the Institute for ComputerSciences, Social-Informatics and Tele-communications Engineering (LNICST). Springer-Verlag. Disponível em: <2011_securecomm.pdf>. , set 2011

NBC. **Facebook Defends its Real Name Policy | NBC Bay Area**. Disponível em: <http://www.nbcbayarea.com/news/business/Facebook-Defends-its-Real-Name-Policy.html>. Acesso em: 29 set. 2011.

O REILLY, T. **What is Web 2.0:** Design patterns and business models for the next generation of software. Communications and Strategies, v. 65, p. 17, 2007.

PAQUIN, C. U-Prove Cryptographic Specification V1. 1 Draft Revision 1, February 2011. Disponível em: **http://www. microsoft. com/u-prove**. Acesso em: 29 set. 2011.

PEYTON, L.; DOSHI, C.; SEGUIN, P. **An audit trail service to enhance privacy compliance in federated identity management**. Proceedings of the 2007 conference of the center for advanced studies on Collaborative research. **Anais**..., CASCON '07. New York, NY, USA: ACM. Disponível em: <http://doi.acm.org/10.1145/1321211.1321230>. , 2007

PFITZMANN, A.; HANSEN, M. **Anonymity, unlinkability, unobservability, pseudonymity, and identity management-a consolidated proposal for terminology**. : Citeseer, 2005.

PFITZMANN, A.; HANSEN, M. **A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management**. Citeseer, 2009.

POINTCHEVAL, D.; STERN, J. **Provably secure blind signature schemes**. Advances in Cryptology—ASIACRYPT'96. **Anais**... . , 1996

RECORDON, D.; FITZPATRICK, B. **OpenID authentication 2.0-final**.

RECORDON, D.; REED, D. **OpenID 2.0: a platform for user-centric identity management**. Proceedings of the second ACM workshop on Digital identity management. **Anais**... , 2006

REITER, M. K.; RUBIN, A. D. **Crowds: anonymity for Web transactions**. ACM Trans. Inf. Syst. Secur., v. 1, n. 1, p. 66–92, nov 1998.

RUDDY, M.; TREVITHICK, P.; NADALIN, T.; OLDS, D. **Higgins trust framework**. Digital ID World, 2006.

SENGUPT, D.; RAJGARHIA, A. **Identity Management in PRPL using OpenID**. http://senguptas.org/Documents/CS294S_OpenID_Final_Project_Report.pdf. Disponível em: <http://senguptas.org/Documents/CS294S_OpenID_Final_Project_Report.pdf>. Acesso em: 10 mar. 2010. , 10 mar 2010

SHMATIKOV, V.; WANG, M.-H. **Timing Analysis in Low-Latency Mix Networks**: Attacks and Defenses. In: GOLLMANN, D.; MEIER, J.; SABELFELD, A. (Eds.). Computer Security – ESORICS 2006. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. v. 4189p. 18-33.

STONE, B.; ALTO, P. **Facebook Aims to Extend Its Reach Across the Web**. The New York Times, v. 12, n. 1, 2008.

SYVERSON, PAUL; CERVESATO, I. **The Logic of Authentication Protocols**. In: FOCARDI, R.; GORRIERI, R. (Eds.). Foundations of Security Analysis and Design. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. v. 2171p. 63-137.

USP. **Anuário Estatístico**. Disponível em: <https://sistemas.usp.br/anuario/>. Acesso em: 24 set. 2011.

VACCA, J. R. **Computer and information security handbook**. Morgan Kaufmann, 2009.

VOSSEN, G.; HAGEMANN, S. Unleashing Web 2.0: From concepts to creativity. **Ubiquity**, v. 2007, p. 3:1–3:1, dez 2007.

WESTIN, A. **Privacy and Freedom**. New York: Atheneum, 1967.

WU, A. **Yahoo! Accepts OpenID Authentication with Google · YDN Blog**. Disponível em: <http://developer.yahoo.com/blogs/ydn/posts/2010/10/yahoo-accepts-openid-authentication-with-google/>. Acesso em: 24 mar. 2011.

ZHANG, F.; SAFAVI-NAINI, R.; SUSILO, W. **Efficient Verifiably Encrypted Signature and Partially Blind Signature from Bilinear Pairings**. In: JOHANSSON, T.; MAITRA, S. (Eds.). Progress in Cryptology - INDOCRYPT 2003. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2003a. v. 2904p. 71-84.

ZHANG, F.; SAFAVI-NAINI, R.; SUSILO, W. **Efficient Verifiably Encrypted Signature and Partially Blind Signature from Bilinear Pairings.** In: JOHANSSON, T.; MAITRA, S. (Eds.). Progress in Cryptology - INDOCRYPT 2003. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003b. v. 2904p. 191-204.