Autenticação com OpenID e Java EE JAAS

Janine Chaves F. Pinto[1], Rafaela Ramos[2]

[1, 2]Instituto de Educação Continuada – Pontifícia Universidade Católica de Minas Gerais (PUCMINAS)
Belo Horizonte – MG – Brazil

nyneferreira@gmail.com, rafaela.ramospr@gmail.com

Abstract. *This paper aims to describe the authentication process using technology and process openID authentication and authorization using JAAS (Java Authentication and Authorization Service) in a university setting. Was used to implement the Java development language.*

Resumo. *Este trabalho tem como objetivo descrever o processo de autenticação utilizando a tecnologia de openID e o processo de autenticação e autorização utilizando o JAAS (Java Authentication and Authorization Service) no cenário de uma universidade. Para a implementação foi utilizado a linguagem de desenvolvimento Java.*

1. Introdução

Atualmente temos uma infinidade de serviços oferecidos na internet e com isso a segurança é uma preocupação crescente para os internautas, que estão sujeitos a diversos golpes. Um golpe muito frequente é o roubo de senhas e logins, o processo de recuperação pode ser bastante complexo para o usuário.

Acredito que a maioria dos usuários possuem muitos identificadores e na maioria das vezes tenta usar o mesmo usuário em todos os sites, porém ele não está disponível em todos os sites. Isso pode levar á uma séria falha de segurança, pois, os usuários costumam a criar senhas fracas para facilitar memorização.Uma forma de solucionar esse problema é utilizar a tecnologia OpenID que oferece a possibilidade de se utilizar um único login e senha para acessar todos os serviços desejados.

Outro fator relevante é a autorização de acesso, ou seja, definir níveis de acesso para as informações disponíveis. Uma solução para este problema é a API (Application Program Interface) JAAS (Java Authentication and Authorization Service) foi desenvolvida pela Sun Microsystems com o propósito de facilitar a utilização dos mecanismos de autenticação.

2. OpenID

OpenID é um protocolo de autenticação descentralizado, ou seja, posso ter apenas um identificador e usá-lo em qualquer site o recurso web que tenha adotado o protocolo. Segundo OpenID Foundation, os principais benefícios em se utilizar OpenID são:

- Acelerar o processo de inscrição: toda vez que vamos criar um login em algum site que não utilize OpenID a quantidade de informações a serem fornecidas é muito grande e repetitiva as vezes, com o OpenID esse processo é desnecessário.

- Reduzir a frustação associados á manutenção de vários nomes de usuário e senhas: com o OpenID é possível utilizar uma única conta existente (a partir dos provedores existentes) para entrar em qualquer site sem precisar criar um login e senha novos, pois usar o mesmo login e senha em vários sites ou serviço é um risco de segurança.
- Obter maior controle sobre a sua identidade na web: com o OpenID você controla quais informações pessoais que deseja compartilhar com os sites.
- Minimizar risco em senhas: como a maioria de usuários usam a mesma senha e usuário para acessar todos os serviços e sites na web e supondo que ocorra um problema de segurança em um desses sites alguém pode ter acesso a sua senha em vários sites. Já com o OpenID as senhas nunca são compartilhadas com todos os sites e se algo ocorrer é so alterar a senha no provedor de OpenID.

A principal vantagem do OpenID é que seu protocolo de autenticação é simples, pois a sua adoção exige poucas alterações do provedor de serviços e nenhuma por parte do usuário.

2.1. Terminologia

A arquitetura do OpenID é composta pelos seguintes agentes:

- Usuário final (End User): é quem deseja acessar um determinado serviço.
- Provedor de Identidades OpenID (OpenID Provider - OP): é o provedor de serviços onde o usuário deseja se conectar.
- Consumidor ou Provedor de Serviços (SP) ou Relying Party (RP): O website onde o usuário deseja conectar-se, utilizando um identificador OpenID. Esse identificador será validado pelo provedor OpenID do usuário.

2.2. Processo de Autenticação

O processo de autenticação utilizando o protocolo OpenID é composto pelos seguintes passos, conforme (Figura 1):

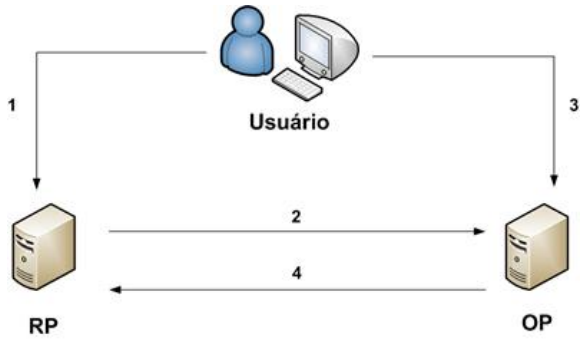

Figura 1. O processo de Autenticação OpenID

1. O usuário conecta-se a um website do provedor de serviços (RP), o RP pede ao usuário se login e senha de seu identificador único OpenID.

2. Em seguida, o RP utiliza o identificador enviado pelo usuário para descobrir qual o provedor OpenID (OP) utilizado para autenticação do usuário. Quando o RP descobre o

provedor utilizado, redireciona o usuário para ele e enviando um pedido de autenticação OpenID.

3. Quando o usuário é redirecionado, uma nova conexão é estabelecida, agora entre usuário e provedor, para realizar a autenticação. O método de autenticação realizado pelo provedor não é especificado pelo protocolo OpenID.

4. Caso as informações inseridas pelo usuário estejam de acordo com sua autenticação, o provedor então informa ao RP que o usuário foi autenticado com sucesso e o redireciona para o RP. Após a validação da confirmação do provedor pelo RP, o usuário poderá ter acesso aos recursos da aplicação web.

2.3. Processo de Descoberta

Segundo Guimarães (2012), o Provedor de Serviços executa o processo de descoberta para identificar qual o provedor utilizado pelo usuário. Existem três formas de descobrimento:

- Se o usuário inserir um XRI: o protocolo XRI será utilizado para obter um documento XRDS (extensible resource desciptor sequence).
- Se o usuário inserir um URI então é utilizados o protocolo Yadis para obter um documento XRD.
- Em ultimo caso deve-se retornar um arquivo HTML contendo os dados do protocolo OpenID.

2.4. Processo de Associação

Conforme Guimarães (2012), o processo de associação é responsável por estabelecer um segredo comum entre o provedor de serviços (SP) e o provedor OpenID, ou seja, apenas os dois conhecem. Esse segredo é utilizado para assinar as mensagens onde, identifica para que a mensagem foi criada pelo provedor e para garantir a integridade de seu conteúdo.

3. Java EE JAAS (Java Authentication and Authorization Service)

De acordo com Filho (2006), o JAAS (Java Authentication and Authorization Service) é uma API (Application Program Interface) de segurança do Java é usado para implementar os mecanismos de autenticação e autorização em uma aplicação Java.

Seu principal objetivo é facilitar a utilização dos mecanismos de autenticação, pois sua utilização desses mecanismos não depende da aplicação Java. Dessa forma quando for preciso modificar o mecanismo de autenticação ou acrescentar um novo mecanismo não é necessário alterar o código fonte da aplicação.

O JAAS pode ser usado para duas finalidades:

- Para autenticação de usuários de forma segura e determinar quem está atualmente acessando a aplicação.
- Para a autorização dos usuários, garantindo que eles tenham os direitos de controle de acesso (permissões) necessários para fazer as ações pertinente a seu nível de acesso.

3.1. Terminologia

Em aplicações Java EE o modelo de segurança mais conhecido é o RBAC(Role Based Access Control), onde papéis são atribuídos para os usuários e cada papel possui determinadas atribuições podendo também atribuir vários papéis para um usuário que pode também possuir uma ou mais permissões (ver figura 2) [Ribeiro, 2009].

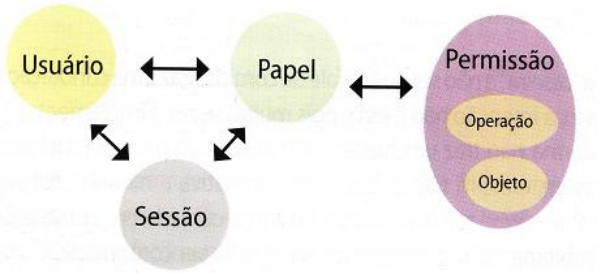

Figura 2. Representação modelo RBAC

Os principais conceitos utilizados em Java EE são:

- Role: é um nível de permissão, usuários podem ter roles (permissões) diferentes e com acessos diferentes.
- User: representa uma entidade com acesso ao sistema. O nível de acesso desse usuário vai depender do seu role (nível de permissão).
- Group: representa os grupos de acesso que pode estar associado com um conjunto de roles e todo usuário que é membro do grupo automaticamente herda os roles associados.
- Realm: é todo o conjunto completo de users, roles e groups normalmente são armazenados em algum banco de dados.
- Constraints: separação de privilégios.

4. Cenário

Este trabalho tem como objetivo o estudo e implementação da tecnologia OpenID e JAAS no cenário de uma universidade, toda a implementação será voltada para o ambiente de desenvolvimento web. Para o desenvolvimento será utilizado linguagem Java, para implementação foi utilizado o ambiente de desenvolvimento Netbeans.

Abaixo descrição dos requisitos contemplados:

- Será disponibilizado um portal para o acesso dos alunos e professores da universidade;

- Os alunos deverão acessar o sistema utilizando sua conta do google.

- Após acesso será visualizada as informações de acordo com o perfil acessado.

4.1. Implementação OpenID

Para implementação do OpenID foi criado um portal para acesso somente dos alunos, não foi implementado o acesso de professores. O aluno acessa o portal e para acessar

sua página deverá efetuar login com sua conta do google. Todo o processo é demonstrado no diagrama de sequência abaixo (ver figura 2).

Figura 3. Fluxo de Autenticação OpenID

1. A aplicação web solicita ao usuário para efetuar o login, oferecendo a opção para uso da conta do google. O usuário seleciona a opção "Entrar com google".
2. A aplicação web realiza o processo de descobrimento enviando uma requisição para o google para obter informações sobre seu processo de autenticação.
3. A aplicação web efetua a associação.
4. A aplicação web envia a requisição de login para o endereço do google, ou seja, o usuário é redirecionado para a página do google para efetuar o login.
5. Após ser efetuado o login, o google exibe uma página de confirmação que notifica o usuário que um aplicativo, está solicitando a autenticação.

Como dito no passo 5 o usuário é informado que um aplicativo está tentando acessar suas informações, está página serve para o usuário confirmar ou rejeitar a solicitação efetuada, o usuário só poderá acessar a aplicação se confirmar o login (ver Figura 3).

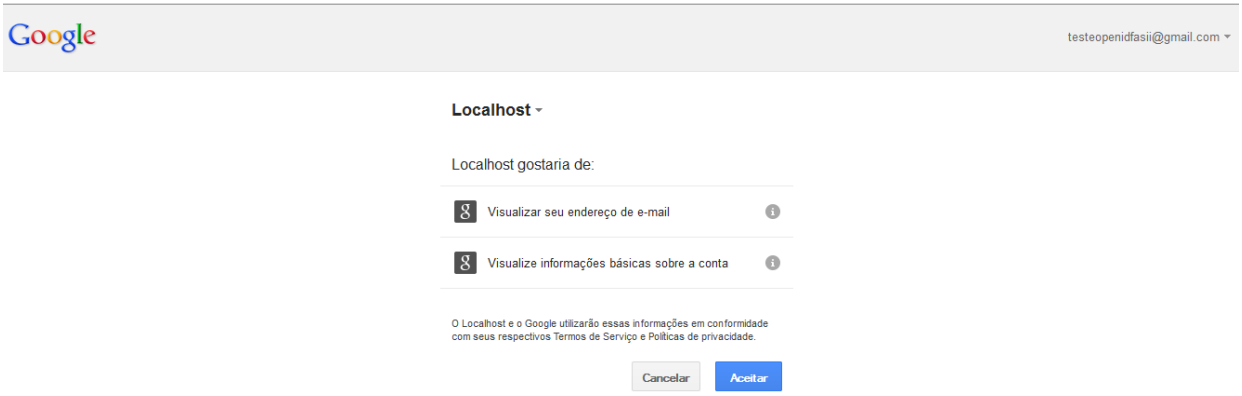

Figura 4. Autorização para acesso Google

4.2. Implementação JAAS

Para implementação de controle de acesso com JAAS foi criado dois perfis para acesso á aplicação que são: aluno e professor. Após o login efetuado com sucesso, cada usuário é redirecionado para a página que tem acesso conforme seu perfil (ver Figura 5).

Figura 5. Tela Inicial

Caso o login seja efetuado com sucesso, ou seja, o usuário está associado a um desses grupos (aluno ou professor) terá acesso a aplicação. Caso contrário será retornada uma página de erro informando ao usuário que ele não tem acesso a aplicação.

Referencias

Guimarães, Pedro Henrique Valverde. (2012) "Arquitetura de Gerenciamento de Identidades Usando OpenID e Cartões Inteligentes", http://www.gta.ufrj.br/ftp/gta/TechReports/Guimaraes12/Guimaraes12.pdf, Outubro.

Ribeiro, Vinicius Gadis. (2009) "Segurança em Aplicações J2EE no Ambiente JBoss", http://www.uniritter.edu.br/graduacao/informatica/sistemas/downloads/tcc2k9/TCCII_Juliano_2009_2.pdf, Outubro.

Silva, André Thiago Souza da., Saldanha, Hugo Vasconcelos. (2009) "Controle de Acesso Baseado em Papéis na Informatização de Processos Judiciais", http://www.cic.unb.br/~rezende/trabs/jurisrbac.pdf, Outubro.

Filho, Paulo Roberto Lelis Goulart. (2006) "Autenticação e Autorização em Java Utilizando JAAS", http://arquivo.ulbrato.br/ensino/43020/artigos/relatorios20052/Arquivos/Paulo%20R%20L%20G%20F%2020Estagio%20Supervisionado%20em%20Sistemas%20de%20Informacao.pdf, Outubro.

OpenID Foundation. "Benefits of OpenID", http://openid.net/get-an-openid/individuals/, Outubro.