Autenticação e Autorização em Arquiteturas Orientadas a Serviço através de Identidades Federadas

Edson T. de Camargo[1] **, Michelle S. Wangham**[12*]**, Joni Fraga**[1*]**, Emerson R. de Mello**[1*]

[1]Departamento de Automação e Sistemas (DAS)
Universidade Federal de Santa Catarina (UFSC) – Florianópolis – SC – Brasil

[2]Grupo de Sistemas Embarcados e Distribuídos–GSED/CTTMAR
Universidade do Vale do Itajaí (UNIVALI) – São José – SC – Brasil

{camargo,fraga,emerson}@das.ufsc.br, wangham@univali.br

Abstract. *The* Single Sign-On (SSO) *authentication enables a user authenticates once in your original domain and uses this authentication in other domains of the distributed system. This article defines a model for authentication and authorization in system oriented services. Using the SAML standard and the identity federated concepts, the model emphasizes the authentication* SSO *between administrative domains with different security infrastructure and the authorization in large-scale environments.*

Resumo. *A autenticação* Single Sign-On (SSO) *permite que um usuário se autentique apenas uma vez em seu domínio de origem e use desta autenticação nos demais domínios de um sistema distribuído. Este artigo define um modelo para a autenticação e a autorização em sistemas orientados a serviços, em ambientes de larga escala. Principalmente através do padrão SAML e do conceito de identidades federadas, o modelo concretiza a autenticação e a autorização considerando domínios administrativos com diferentes tecnologias de segurança, de forma transparente e interoperável.*

1. Introdução

A demanda pelo compartilhamento das informações de forma transparente e segura entre diferentes domínios administrativos e a necessidade de estabelecimento de relações comerciais são exigências cada vez mais atuais, principalmente, no contexto da Internet. Clientes e provedores de serviços são as entidades que se fazem presentes neste domínio e almejam algum tipo de interação. Neste cenário, três barreiras a serem transpostas são: a heterogeneidade das infra-estruturas de segurança, presentes nos diversos domínios corporativos, o estabelecimento de relações de confiança entre entidades desconhecidas e o gerenciamento de identidades feito tanto por provedores de serviços quanto por clientes.

Em sistemas distribuídos, os modelos usuais de autorização se apóiam em uma autoridade de autenticação para mediar a confiança entre partes desconhecidas (terceira parte confiável). Desta forma, as interações entre partes distintas são alcançadas pela apresentação de credenciais emitidas por uma autoridade de autenticação conhecida pelas partes envolvidas. Em ambientes mais complexos como a Internet, este modelo de simples intermediação se apresenta como limitado, já que cada domínio possui suas próprias políticas, infra-estruturas de segurança e ainda possui uma forma particular de

*Bolsista CNPq.

gerenciar as identidades dos principais. Ou seja, cada domínio executa seus controles de autorização a partir de suas políticas locais, sem considerar os atributos dos outros domínios e, geralmente, exigindo antes uma autenticação no próprio domínio.

Abordagens como a autenticação *Single Sign On* (SSO) surgiram, justamente, para tornar mais simples as interações entre clientes e provedores de serviços. O cliente se autentica uma única vez e faz uso dessa autenticação nas interações com os demais provedores de serviços. No entanto, devido a problemas de interoperabilidade, essa abordagem é deficiente em domínios com diferentes infra-estruturas de segurança.

Neste trabalho, é descrito um modelo computacional que apresenta funcionalidades para a transposição de autenticação, considerando domínios administrativos fundamentados em diferentes tecnologias de segurança. Neste modelo, um principal[1] pode acessar recursos em domínios com tecnologias de segurança diferentes do seu domínio de origem, usando para isto as credenciais fornecidas em seu próprio domínio.

Segundo [Jøsang e Pope 2005], no gerenciamento de identidades federadas, cada empresa constrói um domínio, em que estão presentes os seus provedores de serviços, os de identidades e os de credenciais. Os acordos estabelecidos entre domínios podem permitir que identidades locais a um domínio sejam aceitas nos demais domínios que participam do acordo. Desta forma, um usuário com uma identidade registrada em seu domínio pode acessar recursos em outros domínios da federação sem a abertura de um novo registro, uma nova identidade. Neste trabalho, o gerenciamento de identidades federadas contribui de forma significativa para a transposição da autenticação dos principais.

Concretizar a autenticação e a autorização distribuída de forma transparente em sistemas complexos é uma tarefa muito árdua, diretamente afetada pela escalabilidade, e que exige grande cooperação entre domínios administrativos distintos, principalmente, a fim de garantir a interoperabilidade entre os domínios. Para isto, são necessários tanto padrões altamente difundidos, com abstrações suficientes para esconder as diferentes tecnologias, quanto modelos que contribuam para a integração de tais padrões.

A tecnologia de Serviços *Web*, baseada na Arquitetura Orientada a Serviços (AOS), é uma das promissoras soluções para integração de domínios administrativos em sistemas distribuídos [W3C 2004]. Embora esta tecnologia contribua para contornar os desafios envolvendo a autenticação e a autorização em aplicações distribuídas, o grande número de especificações e padrões destinados à segurança tanto impõe um certo nível de complexidade, quanto dificulta sua ampla adoção [Vogels 2004]].

Apoiado nas tecnologias de segurança para Serviços *Web*, este artigo apresenta um modelo para a autenticação e para a autorização em sistemas distribuídos orientados a serviços. Através do conceito de identidades federadas, este modelo concretiza a transposição da autenticação entre domínios administrativos com diferentes tecnologias de segurança (X.509, SPKI/SDSI, por exemplo) de forma transparente e interoperável.

A partir da premissa de que as relações de confiança entre domínios se encontram estabelecidas[2], o modelo provê: (1) a transposição da autenticação, (2) a conseqüente autorização descentralizada, e ainda (3) a transposição de atributos dos clientes. De forma a comprovar a aplicabilidade do modelo, um protótipo foi implementado e integrado a

[1] Usuários, processos ou máquinas autorizados pelas políticas do sistema.

[2] O estabelecimento dinâmico da confiança está sendo abordado em outro trabalho [de Mello e Fraga 2005] no contexto do projeto (CNPQ) Infra-estrutura de Segurança para Aplicações Distribuídas Orientadas a Serviço.

uma aplicação distribuída - um portal de informações voltado para o entretenimento.

O artigo está dividido em seis seções. Nesta primeira seção foram apresentadas a motivação e a justificativa para o desenvolvimento do modelo proposto e os objetivos do artigo. Na próxima seção, uma descrição dos principais aspectos da segurança em serviços Web é apresentada. A transposição da autenticação e a autorização descentralizada oferecidas pelo modelo são detalhadamente descritas na Seção 3 e os aspectos da implementação do modelo são introduzidos na Seção 4. A Seção 5 apresenta uma discussão e comparação com os trabalhos relacionados. Por fim, a última seção apresenta a conclusão do presente artigo.

2. Segurança em Serviços *Web*

A seguir, serão, brevemente, descritas as principais especificações de segurança destinadas à arquitetura dos Serviços *Web* e os padrões de segurança em XML diretamente relacionados ao contexto deste trabalho.

A arquitetura dos Serviços *Web* está ligada ao XML e às extensões de segurança deste padrão definidas pela W3C, tais como ***XML-Signature*** [Bartel et al. 2002] e ***XML-Encryption*** [Imamura et al. 2002]. Estas recomendações permitem expressar assinaturas digitais e cifragem de dados em formato XML, sendo que os dados assinados e/ou cifrados podem ser ou não documentos XML. Estes mecanismos tornam possível a segurança fim-a-fim para os Serviços *Web* que usam o XML para troca e armazenamento de dados.

A **especificação XACML** (*eXtensible Access Control Markup Language*) [OASIS 2005a] descreve tanto uma linguagem para expressar regras de políticas quanto um protocolo pedido/resposta para decisões de controle de acesso. Esta especificação utiliza dois elementos para implantar o controle de acesso em ambientes distribuídos, o PEP (*Policy Enforcement Point*), responsável por mediar e concretizar o acesso, e o PDP (*Policy Decision Point*), que é chamado pelo PEP para efetuar o processamento da política e decidir, com base nas informações do sujeito e do recurso, se o acesso será permitido. Outras duas entidades definidas nesta especificação são o PIP (*Policy Information Point*)e o PAP (*Policy Access Point*). O primeiro é responsável por recuperar informações do sujeito, ambiente e recurso; já o segundo, pelo acesso à política do recurso.

A **especificação SAML** (*Secure Assertion Markup Language*) [OASIS 2005b] é uma infra-estrutura de segurança projetada para expressar informações[3] sobre autenticação, autorização e atributos de um sujeito e que permite a troca dessas informações entre parceiros de negócios. O SAML não provê a autenticação em si, mas sim meios para expressar informações de autenticação. O cliente pode se autenticar apenas uma vez e utilizar dessa autenticação nos demais domínios filiados (*Single Sign On*). Estas características tornam o padrão SAML um importante alicerce para o gerenciamento de identidades federadas no modelo proposto.

A ***WS-Security*** [OASIS 2004b], principal especificação de segurança para Serviços *Web*, apóia-se nos padrões ***XML-Signature*** [Bartel et al. 2002] e ***XML-Encryption*** [Imamura et al. 2002] para prover trocas de mensagens seguras. A especificação visa ser flexível, sendo possível utilizar uma grande variedade de mecanismos de segurança. Mais especificamente, esta tecnologia provê suporte para diferentes tipos de credenciais de segurança[4](*security tokens*), possibilitando que um cliente utilize

[3]Com o objetivo de serem interoperáveis, as informações são expressas em asserções de segurança.

[4]Entre os formatos de credenciais de segurança estão o *UserNameToken*, X.509, Kerberos e SAML.

múltiplos formatos de credenciais para a autenticação e autorização, múltiplos formatos para assinatura e múltiplas tecnologias de cifragem de dados. Estas características são muito importantes para alcançar a interoperabilidade entre tecnologias de segurança de diferentes domínios administrativos.

A **especificação *WS-Policy*** [WS-Policy 2006] provê uma gramática extensível e flexível que possibilita expressar competências, requisitos e características gerais de Serviços *Web*. Define um arcabouço e um modelo para expressar essas propriedades como políticas[5]. A estrutura das asserções de políticas, tais como tipos de credenciais requeridas e algoritmos de cifragem suportados, são definidas na especificação *WS-SecurityPolicy* [WS-SecurityPolicy 2003]. A WS-*Policy* não descreve como essas políticas são divulgadas ou como estas são anexadas a um Serviço *Web*. Os mecanismos para anexar as políticas com elementos XML, WSDL e UDDI são definidos na especificação *WS-PolicyAttachement* [WS-PolicyAttachment 2006].

A especificação ***WS-Trust*** [WS-Trust 2005] é uma complementação da *WS-Security*, que fornece respostas à questão de como duas entidades podem concordar na natureza e nas características das credenciais de segurança. A *WS-Trust* define um modelo de confiança no qual um cliente, que não possui as credenciais solicitadas pelo provedor de serviços, as solicita a uma autoridade que as possui. Tal autoridade é chamada de *Security Token Services* - STS. Este serviço forma a base para o estabelecimento das relações de confiança e é o responsável por emitir, trocar e validar as credenciais. No entanto, essa especificação não aborda como traduzir as informações contidas na credencial de segurança do principal, diante de diferentes tecnologias.

As especificações *WS-Security*, *WS-Trust* e *WS-Policy* fornecem a base para a **especificação *WS-Federation***, que descreve como tais especificações são combinadas para permitir a construção de domínios de confiança. A base para o estabelecimento da confiança na *WS-Federation* é o serviço STS. Porém, a *WS-Federation* adiciona a este as funcionalidades de Provedor de Identidades (IdP) e o serviço de Atributos/Pseudônimos, combinados ou não em uma única entidade. Um provedor de identidade funciona como um serviço de autenticação, nos quais os membros do domínio se autenticam e fazem uso desta autenticação. O serviço de Atributos/Pseudônimos permite proteger a privacidade do usuário, por meio de pseudônimos[6].

A *WS-Federation* se enquadra no modelo centralizado de gerenciamento de identidades federadas, no qual apenas um provedor de identidades e de credenciais é utilizado por todos os provedores de serviços da federação. Neste modelo, um usuário pode acessar todos os serviços presentes na federação utilizando um mesmo identificador. O modelo se assemelha ao modelo de identidade federada, porém com a diferença de não necessitar do mapeamento de credenciais [Jøsang e Pope 2005]. Esta restrição amarra o cliente a um único provedor de identidade da federação. Assim como a *WS-Trust*, a *WS-Federation* não indica como as informações contidas na credencial original devem ser traduzidas diante de tecnologias de segurança distintas.

3. Modelo de Autenticação e Autorização

O modelo proposto neste trabalho, que compõe um conjunto de funcionalidades para a autenticação e para a autorização em sistemas distribuídos, permite que as credenciais de

[5]Introduz meios para expressar políticas de qualidade de serviço relacionadas a segurança e a confiabilidade.

[6]Um identificador opaco randômico, que não é discernível por uma outra entidade.

um usuário, autenticado em seu domínio de origem, atravessem outros domínios administrativos com diferentes tecnologias de segurança, possibilitando a este usuário o acesso a recursos protegidos por outros domínios. Para que a identidade do usuário seja válida nestes domínios, o modelo se apóia no conceito de identidades federadas.

Nesta seção, após a definição dos conceitos de domínio e de identidades federadas, o modelo será primeiramente apresentado em um sentido genérico, com base em uma proposição da IETF [Yavatkar et al. 2000] e da especificação SAML, para na seqüência tomar uma forma mais concreta assumindo as proposições de padrões e tecnologias de Serviços *Web*. O desafio do modelo proposto é lidar com um grande conjunto de especificações de segurança para Serviços *Web* que, em sua maioria, ainda estão em desenvolvimento ou foram recentemente lançadas.

3.1. Domínios de Segurança

No modelo proposto, cada domínio administrativo agrupa clientes e provedores de serviços de acordo com as suas infra-estruturas de segurança subjacentes, como no exemplo da Figura 1. Domínios baseados em diferentes tecnologias, determinam diferentes controles de segurança para proteção de seus recursos. A compatibilidade entre diferentes domínios não é assumida como premissa do modelo, pelo contrário, a transposição da autenticação diante de diferentes tecnologias de segurança é garantida por conceitos introduzidos no modelo. Assume-se como premissa do modelo que os controles de segurança dos domínios seguem modelos tradicionais. Ou seja, clientes devem provar sua identidades junto a uma autoridade de autenticação e, com base nesta autenticação, são verificados os direitos associados para efetivar os acessos a recursos do sistema.

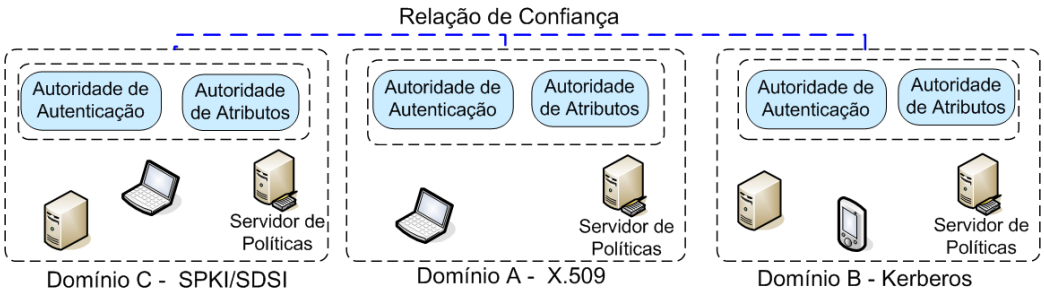

Figura 1. Domínios Federados

Em um domínio de segurança, as autoridades de autenticação, de atributos e o Servidor de políticas são entidades autônomas que gerenciam suas respectivas bases de dados para as consultas realizadas durante as operações de autenticação e de autorização (ver Figura 1). A autoridade de autenticação contém informações de registros de clientes e provedores de serviços do domínio. A autoridade de atributos é responsável por gerenciar os atributos que os usuários informam durante o processo de registro no domínio. Por exemplo *e-mail*, cartão de crédito, CEP, etc. O Servidor de Políticas centraliza as políticas de autorização do domínio, servindo apenas como um repositório.

3.2. Identidades Federadas no Modelo Proposto

As relações de confiança são a base para que as identidades e os atributos de credenciais sejam reconhecidos nos diferentes domínios participantes, possibilitando então a transposição da autenticação. Neste trabalho, parte-se da premissa de que as relações de confiança foram previamente estabelecidas entre as entidades dos domínios.

No contexto de identidades federadas, um cliente pode estar filiado a mais de um domínio, com contas distintas, possuindo diferentes atributos espalhados em diferentes domínios. No modelo proposto, as autoridades de autenticação e de atributos, conhecendo os domínios no qual uma identidade de usuário está federada, podem, a pedido de um provedor de serviços, fazer requisições solicitando atributos do cliente a tais domínios.

Assim como nos trabalhos que implementam o conceito de identidades federadas, tais como *WS-Federation* [WS-Federation 2003], *Shibboleth* [Shibboleth 2005] e *Liberty Alliance*[Liberty 2003], no modelo proposto, o provedor de serviços também é o responsável por recuperar as informações do usuário. O cliente fornece apenas a sua identidade ou pseudônimo e o provedor de serviços é quem procura, nos parceiros da sua federação, os atributos do cliente.

3.3. Autorização no Modelo Proposto

O modelo apresentado neste artigo se baseia no modelo proposto pela IETF [Yavatkar et al. 2000], que descreve dois elementos fundamentais para o governo da política: o PEP (*Policy Enforcement Point*) e o PDP (*Policy Decision Point*) (ver Figura 2). O PEP é o elemento que concretiza o controle de acesso a um recurso, tendo como função aplicar as decisões de políticas. Já o PDP é a entidade que define a política a ser aplicada na requisição de serviço solicitada.

Às proposições da IETF, adicionam-se outras entidades como ponto de partida do modelo: as autoridades de atributos e de autenticação, que participam na autorização de uma requisição (ver 2). Como se assume que as verificações de autorização no modelo são locais aos provedores de serviço, neste caso, tanto o PEP como o PDP devem possuir suas implementações junto aos provedores de serviços que controlam as requisições a estes serviços.

A Figura 2 ilustra o modelo proposto destacando os passos do processo de autorização. Inicialmente, considera-se que os clientes do domínio já foram registrados e que as informações sobre seus atributos estão disponíveis (na autoridade de atributos) aos provedores de serviços, segundo uma política de privacidade definida pelo próprio usuário. Tal política determina quais atributos o usuário deseja informar aos provedores de seu domínio e também a provedores de outros domínios. O cliente se autentica perante a sua autoridade de autenticação (passo 1, Figura 2) e recebe uma asserção declarando sua identidade. Tal asserção declara, por exemplo, que o cliente `cliente@dominoA.com.br` pertence ao domínio e o habilita a solicitar acesso aos recursos dos diversos provedores de serviços localizados nos diferentes domínios.

A concretização de uma política de autorização se inicia no provedor de serviço com o seu PEP (passo 2, Figura 2). Este, ao receber uma requisição de um cliente, emite uma solicitação de decisão de política ao PDP (passo 3, Figura 2). O pedido do PEP para o PDP define um ou mais elementos de política, além de informações sobre o acesso desejado. O PDP, para tomar a decisão referente a requisição de acesso, pode fazer consultas (passo 4) à política de autorização e às autoridades de autenticação e de atributos (passos 6 e 7), a fim de verificar a política de autorização relacionada ao recurso, assim como identificar quem está solicitando o acesso ao recurso e quais são os seus atributos. Após estas verificações (passo 9), o PDP retorna a decisão de política e o PEP a aplica, aceitando ou negando o acesso (passo 10). A justificativa das interações do PDP com as autoridades de autenticação e de atributos do seu domínio é que o provedor de serviços pode não confiar

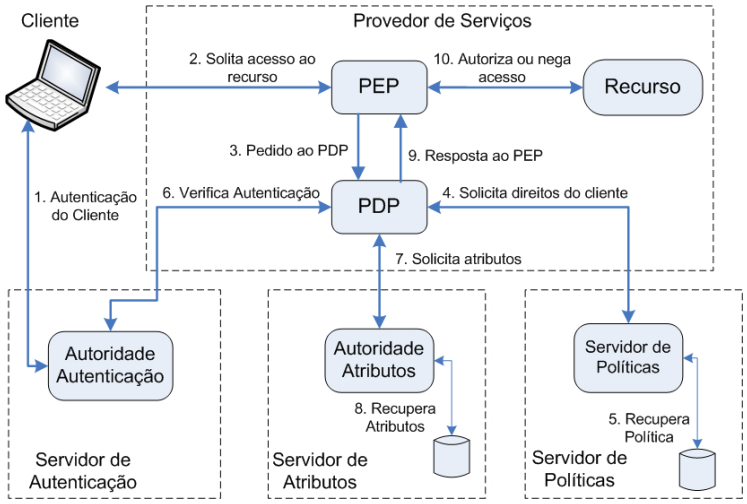

Figura 2. Processo de Autorização no Modelo Proposto

ou desconhecer a autoridade de autenticação do cliente; ou ainda, não suportar o formato da credencial recebida.

3.4. Mapeamento do Modelo em Padrões de Serviços *Web*

O modelo proposto se apóia nos padrões de segurança da tecnologia de Serviços *Web*, assumindo então a forma orientada a serviços, conforme ilustra a Figura 3. As especificações *WS-Trust*, *WS-Federation*, XACML, SAML, *WS-Policy*, *WS-Security*, *XML-Encryption* e *XML-Signature* fornecem os conceitos e os serviços que formam a base da solução proposta.

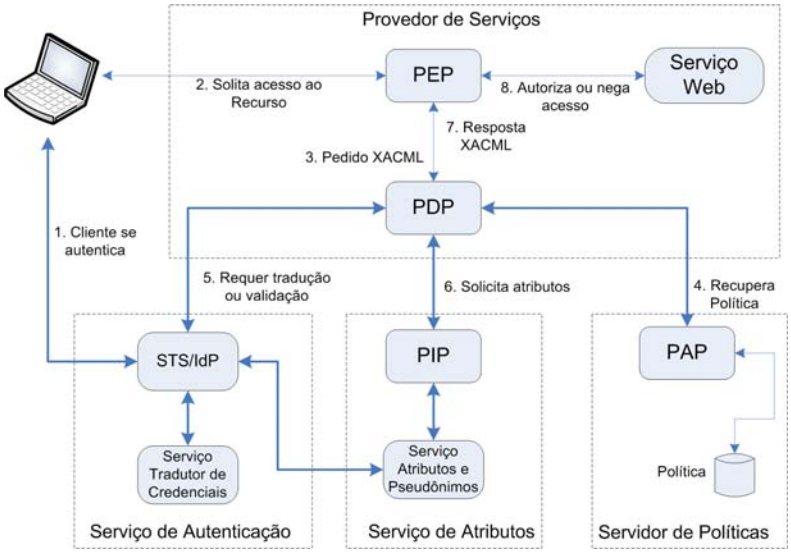

Figura 3. Representação do modelo em Serviços *Web*

O STS (*Security Token Service*), proposto na especificação *WS-Trust*, é o responsável por emitir, validar e trocar credenciais, já o IdP (*Identity Provider*), introduzido na especificação *WS-Federation*, é um tipo especial de STS que visa declarar a identidade do principal em uma credencial. Nesta proposta, ambos são combinados em uma única entidade, chamada STS/IdP. Tanto a privacidade do usuário quanto o fornecimento de atributos são funções do APS (*Attribute and Pseudonym Service*), definido na especificação *WS-*

Federation), que age em conjunto com o STS/IdP. O STS/IdP e o Serviço de Atributos e Pseudônimos agem, respectivamente, em conformidade com a autoridade de autenticação e de atributos definidas no modelo inicial.

Quando as interações entre clientes e provedores de serviços acontecem em um único domínio, não há maiores implicações para a autenticação, uma vez que uma mesma tecnologia de segurança é usada no domínio. Basta o STS/IdP emitir algum atributo de segurança em nome da entidade que deseja acessar ao serviço de um provedor e este, avaliando se os atributos foram emitidos pelo STS que confia, concede acesso ao recurso.

A segurança das mensagens SOAP é garantida através da especificação *WS-Security*, que provê a segurança fim-a-fim e inibe ataques como *replay* e *man-in-the-middle*, utilizando mecanismos para cifragem e assinatura, conforme os requisitos definidos na política da qualidade de proteção expressa segundo a especificação *WS-Policy*[7].

Para o controle de acesso, a solução proposta se apóia no padrão XACML. Quando o PDP necessita realizar requisições por atributos ou validar a credencial apresentada pelo cliente, este o faz através do protocolo definido na especificação *WS-Trust*. Interessante perceber que, no modelo, o STS/IdP age como o PIP do padrão XACML.

3.4.1. A Transposição da Autenticação e o Serviço Tradutor de Credenciais

Os desafios no modelo estão presentes quando uma requisição de acesso envolve a transposição entre domínios distintos. A transposição da autenticação entre domínios tem como base o STS/IdP, o Serviço de Atributos e Pseudônimos (APS), e o Serviço Tradutor de Credenciais (STC), introduzido neste trabalho (Figura 4).

O STS/IdP, através de asserções SAML, assume no modelo o papel de mediador das relações de confiança envolvendo diferentes domínios de segurança. Para atingir a interoperabilidade entre domínios diferentes e transpor as informações de autenticação e de atributos, faz-se neste trabalho largo uso das asserções SAML. O STS/IdP de um domínio emite estas asserções que podem transportar informações de autenticação e de atributos dos principais, mantendo a privacidade do principal. Assume-se que ao se autenticar no STS/IdP do seu domínio, o cliente recebe uma asserção SAML de autenticação, que pode ser usada em qualquer requisição de serviço nos domínios federados.

Uma vez que a especificação *WS-Federation* define um gerenciamento centralizado de identidades, uma extensão ao Serviço de Atributos e Pseudônimos é introduzida no modelo proposto para permitir que este serviço busque os atributos do usuário nos domínios em que o mesmo está filiado (identidade federada). Desta forma, ao receber uma requisição por atributos (passo 6 Figura 3), e não possuindo os atributos solicitados, o Serviço de Atributos e Pseudônimos busca os mesmos nos domínios federados.

Conforme a definição de domínio, as características de um STS/IdP dependem da tecnologia de segurança subjacente. As especificações *WS-Trust* e *WS-Federation* não esclarecem como ocorre a tradução de uma credencial de uma determinada tecnologia para outra credencial de tecnologia diferente. A *WS-Trust* coloca apenas a necessidade de troca de credenciais, já a *WS-Federation* afirma apenas que o provedor de serviços usa o STS para entender e validar as credenciais recebidas do cliente. Nestas especificações,

[7]Vale ressaltar que a segurança de informações sensíveis armazenadas nos clientes está fora do escopo deste trabalho.

mesmo que o STS entenda duas tecnologias de segurança, como X.509 e SPKI, não é dito como as informações do cliente podem ser traduzidas de X.509 para SPKI.

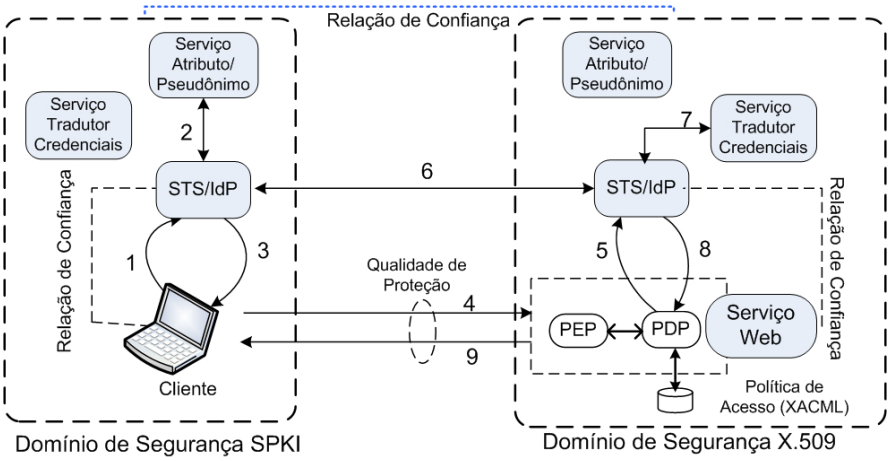

Figura 4. Transposição da Autenticação

No modelo proposto, o Serviço Tradutor de Credenciais (STC) realiza a tradução das informações presentes na asserção SAML para as necessidades da tecnologia subjacente (ver Figura 4, passo 7). Este serviço entende a tecnologia do domínio e é capaz de tanto representar informações de outras tecnologias em uma asserção SAML quanto em informação entendida pela tecnologia do domínio. Desta forma, clientes e provedores de serviço mantém suas características de segurança e o STS/IdP, APS e o STS fazem a ligação entre estes. O processo de tradução consiste, basicamente, em gerar um novo certificado com a chave pública do cliente contida na asserção SAML.

Conforme apresentado na Figura 4, o cliente se autentica em seu STS/IdP e recebe deste uma asserção SAML de autenticação que representa suas informações (passos 1 a 3) e a apresenta ao provedor de serviços (passo 4). A asserção de autenticação contém a chave pública do cliente e a assinatura do STS do cliente (e ainda outras informações que o STS do cliente poderia incluir nessa asserção). O provedor de serviço então repassa a asserção SAML ao STC (passo 5).

O Serviço Tradutor de Credenciais (STC) do provedor de serviço, ao receber a asserção de autenticação (passo 7), retira a chave pública do cliente desta asserção e monta um novo certificado em nome desse cliente. O certificado gerado obedece a tecnologia de segurança subjacente do domínio do STC. A versão atual do STC entende asserções SAML, certificados X.509 [Housley et al. 2002] e certificados SPKI [Ellison 1999]. Em um domínio SPKI, o STS/IdP solicita que o STC monte um certificado de delegação SPKI com a chave pública do cliente extraída da asserção SAML.

De forma semelhante, para um domínio X.509, o STC retira da asserção as informações do cliente (como, por exemplo, a sua chave pública e o instante em que o cliente se autenticou) e as devolve ao STS/IdP. O STS/IdP, agindo como uma autoridade certificadora, emite um certificado X.509 ao cliente e o repassa ao provedor de serviços.

3.4.2. Solicitação de atributos e o Serviço de Atributos e de Pseudônimos (SAP)

No modelo, é o provedor de serviços quem busca as informações do cliente, tornando o acesso mais fácil e transparente ao cliente. Sendo assim, o cliente não precisa fornecer

diretamente seus atributos ao provedor de serviços.

Conforme ilustra a Figura 4, o cliente se autentica normalmente em seu STS/IdP (passo 1), obtém um pseudônimo do SAP (passo 2), recebe a asserção de autenticação (passo 3) e realiza a invocação ao serviço (passo 4). O serviço então pode recorrer a seu STS/IdP para recuperar atributos deste cliente (passo 5). O seu STS/IdP então interage com o STS/IdP do cliente para recuperar os atributos do cliente (passo 6). Uma vez que os atributos solicitados obedecem a uma sintaxe definida para os domínios, os atributos são retornados em uma asserção SAML de atributos. O serviço de atributos e de pseudônimos proposto garante ainda proteção da privacidade da identidade do usuário, caso provedores de serviços formem um conluio para tentar rastrear as preferências de seus usuários.

4. Implementação

Um protótipo envolvendo o modelo proposto neste trabalho foi definido e implementado visando comprovar a sua flexibilidade, bem como a viabilidade de sua utilização em aplicações distribuídas baseadas em uma arquitetura orientada a serviços.

A Figura 5 ilustra a arquitetura do protótipo. Para o desenvolvimento de aplicações baseadas na arquitetura dos Serviços *Web* (camada de transporte), escolheu-se o *framework* de código aberto *Apache Axis*. Para compor a camada de segurança que provê a qualidade de proteção e fornece mecanismos de segurança às mensagens trocadas entre clientes e provedores de serviços, adotou-se as implementações da biblioteca *WSS4J*[8] e a biblioteca *XML-Security*[9], todas da *Apache Software Fundation*. Também foram utilizadas as bibliotecas de código aberto *SunXACML*[10], *JSDSI* implementada por [Morcos 1998] e *Libsdsi*, esta última desenvolvida dentro do projeto cadeias de confiança[11]. As asserções SAML são implementadas por meio da biblioteca *OpenSAML*

Na camada de aplicação, três Serviços *Web* são necessários para prover as funcionalidades do modelo: o STS/IdP (*Security Token Service/Identity Provider*), o Serviço de Atributos e de Pseudônimos (SAP) e o Serviço Tradutor de Credenciais (STC). Ainda na camada de aplicação, estão localizados os clientes e os provedores de Serviços *Web* (ver Figura5).

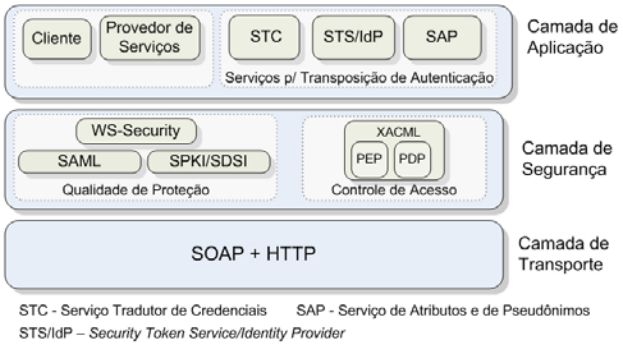

Figura 5. Arquitetura do protótipo

De forma a suportar a emissão de asserções no formato SAML (*SAMLTokens*[OASIS 2004a]), estendeu-se o STS da biblioteca WSS4J, já que este somente emite

[8]Esta biblioteca implementa a especificação *WS-Security* - http://ws.apache.org/wss4j/ - versão 1.0

[9]http://xml.apache.org/security/ - que contém o *XML-Signature* e o *XML-Encryption*

[10]Implementação da especificação XACML - http://sunxacml.sourceforge.net/

[11]http://www.das.ufsc.br/seguranca

tokens X.509. Além disso, adicionou-se a esta biblioteca a capacidade de assinar e cifrar mensagens SOAP utilizando a infra-estrutura SPKI, através das bibliotecas JSDSI.

Um portal de entretenimento, inicialmente apresentado em [de Mello et al. 2006], foi integrado ao protótipo implementado. O objetivo do portal é reunir em uma única interface diversos provedores de serviços que tenham como área de atuação o entretenimento pessoal, como por exemplo, cinemas, parques de diversão, vídeo locadoras, teatros, etc. Um cliente, para acessar os serviços disponibilizados via Portal, deve se autenticar fornecendo um login e senha válidos, sobre uma sessão SSL, conforme ilustrado na Figura 6. Tal portal, em nome do cliente, solicita ao seu STS uma asserção de autenticação SAML. Vale ressaltar que a tecnologia de segurança suportada no portal é o SPKI. Deste ponto em diante, serão adicionadas a todas as requisições deste cliente a asserção de autenticação SAML emitida pelo STS do Portal, para que assim o cliente possa obter acesso aos recursos fornecidos pelos provedores de serviços que podem estar em um domínio X.509 ou SPKI/SDSI (ver Figura 6). Vale lembrar que ao receber a asserção SAML o provedor de serviços solicita ao seu STS que realize a tradução da asserção para um certificado X.509[12].

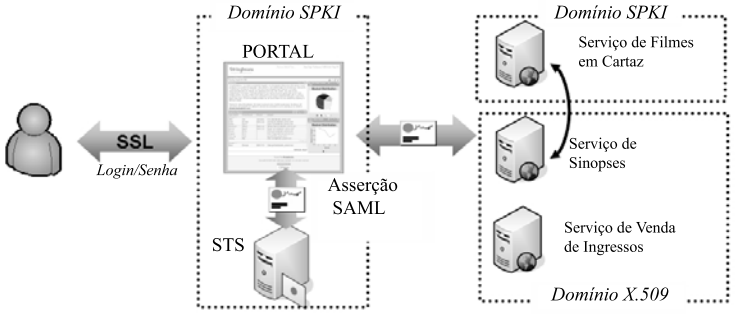

Figura 6. Portal de Entretenimento

5. Trabalhos relacionados

Na literatura, alguns trabalhos se esforçam em reunir um conjunto de especificações relativas a Serviços *Web* para efetuar a transposição de autenticação. Entre estes, a maioria envolve o conceito de identidades federadas através de Serviços *Web*, porém consideram apenas o uso do X.509 como infra-estrutura de segurança [Vecchio et al. 2005, Shibboleth 2005, Kafura et al. 2003]. Não há clareza sobre como ocorre a interação entre domínios com diferentes infra-estruturas de segurança.

O CredEx [Vecchio et al. 2005] facilita o gerenciamento de credenciais por parte do usuário através de um repositório centralizado. O modelo se propõe a armazenar credenciais de várias tecnologias de segurança em um repositório e as usa de acordo com os serviços invocados e com suas respectivas tecnologias de segurança. No entanto, a implementação apenas considera a associação prévia de um usuário e senha a um certificado X.509 *proxy* ou vice-versa. Sendo assim, não lida com diferentes tecnologias de segurança, como SPKI ou Kerberos. Também não permite que credenciais sejam traduzidas dinamicamente. No modelo aqui definido, há a conversão dinâmica da credencial pelo provedor de serviços, que permite ao cliente manter seus atributos completamente

[12]Mais informações sobre a implementação do protótipo e a integração ao portal estão disponíveis em http://www.das.ufsc.br/seguranca/webservices/

independente das tecnologias de segurança. Futuramente, através do Serviço Tradutor de Credenciais, será possível ainda que, além do X.509 e do SPKI/SDSI, credenciais segundo o formato Kerberos possam ser traduzidas, permitindo a comunicação com provedores e clientes que suportem esta tecnologia.

A especificação *WS-Federation* [WS-Federation 2003] fornece os serviços de STS/IdP e Serviço de Atributos e Pseudônimos para as identidades federadas introduzidas no modelo apresentado. Porém, esta mesma especificação considera que os atributos do usuário são centralizados nos serviços descritos. O trabalho aqui proposto estende esse modelo para permitir a busca nos domínios em que o usuário tem a sua identidade reconhecida como federada, o que permite ao usuário ser referenciado com mais de um identificador ou mesmo pseudônimos. O projeto *InfoCard*, da *Microsoft*, prevê tal extensão da *Ws-Federation*, mas não faz uso de asserções SAML para conduzir os atributos do usuário e sim de uma credencial de transferência chamada *InfoCard*. A *WS-Federation* também não esclarece como se dá a troca de credenciais entre domínios distintos.

O trabalho discutido em [Dyke 2004] apóia-se em especificações como *WS-Trust* e *WS-Federation* e propõe uma arquitetura para construção dinâmica de redes de confiança entre Serviços *Web* no contexto de serviços médicos. O trabalho se preocupa principalmente com o estabelecimento dinâmico da confiança. Para tanto, o autor propõe níveis de confiança (*trust levels*), grupos de confiança (*trust groups*) e autoridades de confiança (*trust authorities*) para auxiliar o STS na emissão de uma credencial a determinado cliente. As relações dinâmicas são criadas após a avaliação dos níveis e grupos de confiança, segundo a política de segurança do serviço. No entanto, o trabalho não se preocupa com outras relevantes questões, tais como lidar com diversas tecnologias de segurança presentes na federação e nem com o processo de autorização. Além disso, incluir informações na credencial para transportar os níveis e grupos de confiança compromete a interoperabilidade, uma vez que o mesmo não define uma sintaxe para tanto. O trabalho também não explora o potencial das asserções SAML, que naturalmente incluem informações sobre a autenticação do usuário e são extensíveis.

O Projeto Shibboleth [Shibboleth 2005] está baseado no conceito de federações e permite também a transposição de autenticação. Também prevê a troca de atributos através de asserções SAML, fazendo o uso de uma padronização dos mesmos. Há uma grande preocupação do Shibboleth quanto à privacidade e anonimato dos usuários. A maioria das organizações que adotam o Shibboleth fazem uso do padrão X.509 para autenticação dos servidores e uma autenticação baseada em senha para autenticação dos clientes. O modelo proposto neste trabalho difere do Shibboleth por dar mais liberdade aos membros da federação, já que estes não precisam fazer uso de um navegador *Web* em suas interações com serviços. O modelo proposto permite também a comunicação direta entre provedores de serviços, ou seja, provedor a provedor. Provedores de serviços não precisam seguir uma rígida padronização para fazer parte da federação, basta estarem associados a um STS/IdP. Além disso, também podem permanecer com sua tecnologia de segurança e fazer uso do Tradutor de Credenciais quando há a necessidade de lidar com diferentes tecnologias.

Cardea [Kafura et al. 2003] é um sistema de autorização distribuído desenvolvido como parte do *Grid* Computacional de Informações da NASA (NASA *Information Power Grid*) [Foster et al. 1998, Foster e Kesselman 1998]. Construído sobre padrões como SAML, XACML e *XMLSignature*, o sistema avalia dinamicamente os pedidos de

autorização, considerando as características dos recursos e do solicitante ao invés de avaliar apenas identidades locais. Os usuários são identificados por certificados X.509 *proxy*. As informações necessárias para completar uma decisão de autorização são avaliadas e coletadas durante o processo de autorização. Essa informação é recolhida de forma apropriada e apresentada ao PDP, que retorna a decisão de autorização final para o pedido, junto com qualquer detalhe relevante. Ao contrário do trabalho aqui proposto, o Cadea constrói a federação através do padrão SAML, ou seja, autoridades SAML, e não através das especificações *WS-Trust* e *WS-Federation*, além disso não considera diferentes tecnologias de segurança envolvidas no processo de autenticação.

6. Conclusão

O conceito de identidades federadas, no qual o trabalho é constituído, (1) favorece uma eficaz e independente transposição, (2) permite ao cliente utilizar os recursos da federação perante a autenticação realizada no seu domínio e (3) aumenta, potencialmente, a quantidade de possíveis clientes dos provedores de serviços.

Por meio das definições propostas no modelo, um cliente desconhecido ao provedor de serviços pode ter sua identidade autenticada graças à confiança estabelecida entre os domínios. O provedor não precisa ter conhecimento prévio dos seus potenciais clientes, facilitando assim a utilização do serviço por diversos clientes e aumentando as possibilidades de negócios entre provedores de serviços.

O modelo apresentado contornou as dificuldades da transposição propondo o Serviço Tradutor de Credenciais para traduzir as credenciais não entendidas pelo provedor. Além disso, reuniram-se diversas especificações de segurança em Serviços Web na solução proposta, tarefa não trivial devido à complexidade e quantidade destas.

Quanto a escalabilidade do modelo proposto, uma vez que cada domínio possui seu próprio provedor de credenciais, a comunicação entre os domínios é suficiente para concretizar a troca de credenciais. É certo que tal dinâmica não apresenta problemas de escala, já que é isto o que ocorre nas atuais aplicações da Internet. Ou seja, no modelo proposto, não há uma entidade centralizadora responsável pelo mapeamento das credenciais. Cada domínio fica responsável por fazer os mapeamentos necessários e assim sendo a escala do sistema está garantida, bastando que domínios bem requisitados provenham soluções para distribuição de carga através, por exemplo, de aglomerados de máquinas.

Agradecimentos

Este trabalho está sendo desenvolvido no contexto do projeto “Infra-estrutura de Segurança para Aplicações Distribuídas Orientadas a Serviço”, financiado pelo CNPq (550114/2005-0). Os autores agradecem aos membros deste projeto por suas contribuições e ao CNPq pelo suporte financeiro.

Referências

Bartel, M., Boyer, J., e Fox, B. (2002). *XML-Signature Syntax and Processing*. W3C.

de Mello, E. R. e Fraga, J. (2005). Mediation of trust across web services. In *ICWS*, pages 515–522. IEEE Computer Society.

de Mello, E. R., Wangham, M. S., Fraga, J., e Camargo, E. (2006). *Livro dos Minicursos do VI Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais*, chapter Seg. em Serviços Web, pages 1–48.

Dyke, J. W. V. (2004). Establishing federated trust networks among web services.

Ellison, C. M. (1999). *RFC 2692: SPKI requirements*. The Internet Society.

Foster, I. e Kesselman, C. (1998). Globus: A toolkit-based grid arch. In *The Grid: Blueprint for a Future Comp. Infrast.*, pages 259–278. MORGAN-KAUFMANN.

Foster, I., Kesselman, C., Tsudik, G., e Tuecke, S. (1998). A security architecture for computational grids. In *Proceedings of the 5th ACM Conference on Computer and Communications Security (CCS-98)*, pages 83–92, New York. ACM Press.

Housley, R., Polk, W., Ford, W., e Solo, D. (2002). *Internet X. 509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. IETF RFC 3280.

Imamura, T., Dillaway, B., e Simon, E. (2002). *XML Encryp. Syntax and Proc.* W3C.

Jøsang, A. e Pope, S. (2005). User centric identity management. In *AusCERT Asia Pacific Information Technology Security Conference 2005*.

Kafura, D., Lorch, M., Lepro, R., Proctor, S., e Shah, S. (2003). First experiences using xacml for access control in.

Liberty (2003). *Introd. to the Liberty Alliance Identity Architecture*. Liberty Alliance.

Morcos, A. (1998). *A Java implementation of SDSI*. Dissertação de mestrado, MIT, MIT.

OASIS (2004a). *Web Services Security: SAML Token Profile*. Organization for the Advancement of Structured Information Standards (OASIS).

OASIS (2004b). *WS Security: SOAP Message Security 1.0*. OASIS.

OASIS (2005a). *eXtensible Access Control Markup Language (XACML) version 2.0*. Organization for the Advancement of Structured Information Standards (OASIS).

OASIS (2005b). *Security Assertion Markup Language (SAML) 2.0 Technical Overview*. Organization for the Advancement of Structured Information Standards (OASIS).

Shibboleth (2005). *Shibboleth Architecture*.

Vecchio, D. D., Humphrey, M., Basney, J., e Nagaratnam, N. (2005). Credex: User-centric credential management for grid and web services. In *ICWS*, pages 149–156.

Vogels, W. (2004). Ws are not distrib. objects. In *Int. CMG Conference*, pages 317–324.

W3C (2004). *Web Services Architecture*. `http://www.w3.org/TR/ws-arch/`.

WS-Federation (2003). *Web Services Federation Language*.

WS-Policy (2006). *Web Services Policy Framework*.

WS-PolicyAttachment (2006). *Web Services Policy Attachment (WS-PolicyAttachment)*.

WS-SecurityPolicy (2003). *Web Services Security Policy Language(WS-SecurityPolicy)*.

WS-Trust (2005). *Web Services Trust Language (WS-Trust)*.

Yavatkar, R., Pendarakis, D., e Guerin, R. (2000). *A Framework for Policy-based Admission Control*. Internet Engineering Task Force RFC 2753.