Um Modelo para o Gerenciamento Federado do SPKI/SDSI através do Serviço XKMS

Michelle S. Wangham*, Joni da Silva Fraga*, Emerson Ribeiro de Mello*, Josiane Milanez

[1]Departamento de Automação e Sistemas – Universidade Federal de Santa Catarina
Campus Universitário, C.P. 476 – CEP 88040-900 Florianópolis, SC

{wangham, fraga, emerson, josiane}@das.ufsc.br

Abstract. *The XML Key Management Specification (XKMS) moves the complexity associated with Public Key Infrastructure (PKI) to a trusted Web Service. Although the specification shows that is possible to use PGP or SPKI/SDSI it is straight focused in X.509 PKI. This work does use of XKMS to propose a federated management model for SPKI/SDSI which permits that distributed applications can get the advantages of an authentication and authorization descentralized model.*

Resumo. *O propósito do XML Key Management Specification (XKMS) é facilitar o uso de uma Infra-estrutura de Chave Pública (ICP), transferindo a complexidade da mesma para um Serviço Web de confiança. Apesar de trazer indicações de como se adequar a ICPs como o PGP e o SPKI/SDSI, esta especificação está fortemente focada na ICP X.509. Para que aplicações distribuídas possam usufruir das vantagens de um modelo descentralizado de autenticação e de autorização, como o do SPKI/SDSI, este trabalho define um modelo de gerenciamento federado através do XKMS.*

1. Introdução

Diante da necessidade da interação entre as aplicações distribuídas de diferentes organizações, que geralmente não foram desenvolvidas para serem interoperáveis, uma nova tecnologia para sistemas distribuídos surgiu, possibilitando assim a troca de informações e a integração com os sistemas legados existentes - os Serviços *Web*.

Os Serviços *Web* seguem uma arquitetura orientada a serviços (AOS) e as principais características que os tornam uma tecnologia emergente e promissora são: (1) possuem um modelo fracamente acoplado e transparente que garante a interoperabilidade entre os serviços, sem que estes necessitem ter o conhecimento prévio de quais tecnologias subjacentes estão presentes em cada lado da comunicação; (2) são auto-contidos e auto-descritivos, e (3) usam padrões abertos como o HTTP e o XML, permitindo assim que aplicações sejam integradas através de linguagens e protocolos amplamente aceitos.

Vários padrões de segurança, tais como o *XMLEncryption* [Imamura et al. 2002], o *XMLSignature* [Bartel et al. 2002] e o SAML [OASIS 2005], são utilizados para prover segurança às informações expressas em XML e como conseqüência aos Serviços *Web*.

*Bolsista CNPq

Estes padrões estão fundamentados no uso de infra-estrutura de chaves públicas (ICPs), através da manipulação de chaves e da emissão de certificados. Entre as ICPs suportadas nestes padrões, destacam-se o X.509, o PGP e o SPKI/SDSI.

O *XML Key Management Specification* (XKMS) [Hallam-Baker e Mysore 2005] é uma especificação aberta que define interfaces, baseadas em Serviços *Web*, que visa retirar dos desenvolvedores de aplicações a complexidade em se trabalhar com uma ICP. Quando se transfere o gerenciamento de uma ICP para o serviço XKMS, as informações e operações desta infra-estrutura ficam acessíveis através de um protocolo padrão independente da tecnologia de segurança subjacente.

O Serviço XKMS está fortemente focado na ICP X.509, que apesar de ser amplamente usada, seu modelo de confiança global e hierárquico impõe algumas restrições quanto a escalabilidade e a flexibilidade. Visto ainda, que a imposição hierárquica de confiança esbarra em problemas da legislação de alguns países, entre outros.

As abordagens que seguem um modelo descentralizado, como o SPKI/SDSI (*Simple Public Key Infrastructure - Simple Distributed Security Infrastructure*) [Ellison et al. 1999, Rivest e Lampson 1996], não apresentam os problemas citados acima. Em [Santin 2004], foi proposta uma extensão do modelo de confiança do SPKI/SDSI, chamada Federação SPKI, que facilita a publicação e a localização de certificados de nome e de autorização, fornecendo também suporte para a criação de novas cadeias de autorização. O modelo de gerência proposto em [Santin 2004], visa preservar a filosofia adotada no modelo SPKI/SDSI, na qual o principal, que deseja obter acesso a algum recurso, é inteiramente responsável pela busca das cadeias de certificados que lhe forneçam o direito de acesso ao recurso. No entanto, esta filosofia sobrecarrega o cliente, já que este necessita entender a complexidade da ICP, além de ser o responsável pela busca de certificados na teia de federações.

Este artigo tem por objetivo descrever uma extensão ao modelo de Federações SPKI [Santin 2004] através do uso de serviços XKMS. No modelo proposto neste artigo, a responsabilidade pela localização das cadeias de autorização é transferida para o serviço XKMS, que também passa a ser o responsável pelo gerenciamento de confiança entre os membros e associados da Federação SPKI. Além do modelo de gerenciamento federado para o SPKI/SDSI através do XKMS, este trabalho apresenta ainda um algoritmo flexível para localização de cadeias de autorização. De forma a comprovar a sua aplicabilidade, um protótipo do modelo foi implementado e integrado a uma aplicação distribuída.

2. Modelo de Confiança SPKI/SDSI e as Federações SPKI

O SPKI/SDSI segue um modelo igualitário, em que os sujeitos (ou principais) são chaves públicas e cada chave pública é uma entidade certificadora [Clarke 2001]. Neste modelo, não há uma entidade centralizadora para o registro de chaves públicas e emissão de certificados, como a autoridade certificadora do X.509, e tão pouco uma infra-estrutura global hierárquica. O emissor de um certificado de autorização SPKI pode permitir que o sujeito delegue as permissões recebidas a outros principais, construindo assim uma cadeia de certificados de autorização [Ellison et al. 1999, Rivest e Lampson 1996]. O modelo de delegação do SPKI/SDSI permite construir cadeias de confiança que partem de um provedor de serviço e que terminam em chaves de clientes deste serviço.

Uma limitação deste modelo está em identificar, entre os certificados de um cli-

ente, caminhos de confiança que o levem ao servidor desejado. Em alguns casos, um cliente e um serviço podem não estar conectados por uma cadeia de confiança. Em [Santin 2004], são propostas extensões para o modelo de confiança do SPKI/SDSI que contornam esta limitação. As Federações SPKI, que agrupam principais com interesses comuns, atuam como agentes facilitadores na localização de certificados e de principais. Suas principais funções estão centradas em seu gerente de certificados que provê mecanismos de armazenamento, de recuperação e de criação de cadeias de autorização. Através do compartilhamento dos certificados de nomes e de autorização em repositórios, os clientes passam a ter uma alternativa a recorrer diante da falta de cadeias apropriadas para o acesso desejado. Nesta forma de organização, um membro de uma federação pode participar de outras federações e diferentes federações podem ter vínculos entre si (relações de confiança), formando assim teias de federações de escopo global. A teia de federação ajuda um cliente na busca de privilégios de acesso que o liguem ao serviço desejado.

3. Serviço XKMS

A especificação XKMS [Hallam-Baker e Mysore 2005] define um Serviço *Web* para a gerência de ICPs que, através de protocolos independentes da tecnologia de segurança subjacente, viabiliza às aplicações distribuídas um maior foco nas atividades de negócios, deixando a complexidade de gerenciamento e o manuseio da ICP sob a responsabilidade de um Serviço *Web*. Com o uso do serviço XKMS, diferentes ICPs podem ser utilizadas sem a necessidade de modificar a aplicação distribuída.

Na especificação XKMS, são definidos protocolos que possibilitam: (1) a geração de pares de chaves; (2) o armazenamento, a localização e a validação de informações de chaves públicas, e também, (3) a validação de assinaturas. Esta especificação se divide em duas partes: o X-KISS (*XML Key Information Service Specification*), que tem por objetivo localizar as informações associadas com as chaves, e o X-KRSS (*XML Key Registration Service Specification*), responsável pelo registro dessas informações.

O principal objetivo do X-KISS é auxiliar as aplicações a trabalharem com assinaturas expressas no padrão *XML Signature* [Bartel et al. 2002]. Na especificação do *XML Signature*, é descrito que um emissor de assinaturas pode incluir informações adicionais sobre a chave pública utilizada para criar a assinatura, fornecendo assim, meios para que o receptor possa verificar a validade da mesma. O XKISS define duas operações: uma para permitir a localização de informações relacionadas às chaves (*Locate*) e outra para verificar se estas informações relacionadas são válidas (*Validate*). As informações relacionadas a um elemento `<ds:KeyInfo>`, proveniente de uma assinatura, podem ser obtidas em uma base local ou através do encaminhamento do pedido a outros serviços XKMS.

O XKRSS permite que um cliente associe informações a uma chave. Essas informações podem ser: um nome, um identificador (p.ex., e-mail) ou ainda atributos específicos para certos tipos de aplicações. O gerenciamento destas informações associadas à chave pública é também realizado através deste mecanismo. O XKRSS define os seguintes serviços [Hallam-Baker e Mysore 2005]:

- registrar: associa informações a uma chave;
- recuperar: permite recuperar a chave privada associada anteriormente e gerada pelo XKRSS;

- reemitir: permite gerar novas credenciais na ICP subjacente (reemissão das informações de uma chave);
- revogar: revoga as informações associadas a uma chave.

4. Um Modelo de Gerenciamento Federado através do XKMS

Segundo Santin [Santin 2004], o modelo de gerência de confiança do SPKI/SDSI define políticas de autorização distribuídas através de cadeias de confiança que são construídas através da delegação de certificados de autorização. As cadeias são caminhos não controlados, sendo da responsabilidade dos membros das mesmas a armazenagem e a recuperação das seqüências de certificados para resolver nomes e/ou para comprovar autorizações. Este modelo está fortemente focado no cliente que deseja acessar a um recurso.

Com intuito de abstrair das aplicações a complexidade do uso e gerenciamento das ICPs, o modelo proposto neste trabalho está baseado no uso do XKMS que atribui estas responsabilidades a um Serviço *Web*. O modelo de gerenciamento federado proposto é uma extensão do modelo de federação SPKI, sendo que as funções de gerenciamento e estabelecimento de confiança entre os elementos da federação, antes atribuídas ao gerente de certificados, são repassadas ao serviço XKMS. Além disso, visando tornar as aplicações clientes independentes da tecnologia de segurança subjacente, a responsabilidade pela localização das cadeias de certificados passa também a ser assumida pelo serviço XKMS.

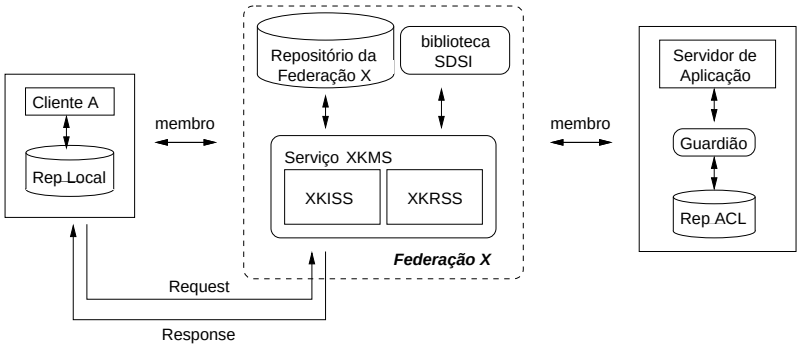

Figura 1. Serviço XKMS e os Elementos da Federação SPKI

Desta forma, neste novo modelo, o serviço XKMS facilita a interação entre o cliente e o servidor de aplicação, encapsulando as funções de gerenciamento de uma federação SPKI e provendo o suporte para navegar na teia de federações, sem caracterizar-se como uma chave intermediária, conforme ilustrado na Figura 1. O serviço XKMS de cada federação tem acesso ao repositório da sua federação e provê dois serviços: o XKISS e o XKRSS (ver Figura 1). O XKISS é o responsável pelo gerenciamento das informações SPKI e oferece as operações de localização e validação. Já o XKRSS é o usado no gerenciamento de uma federação SPKI no oferecimento da operação de registro[1].

Apesar da especificação XKMS dar suporte a diferentes ICPs, a integração do serviço XKMS ao modelo de confiança SPKI não é simples, pois esta especificação

[1] As operações de reemissão e recuperação do XKRSS não são utilizadas no modelo proposto, pois estas são específicas para o gerenciamento de chaves privadas e na filosofia SPKI somente o detentor da chave privada pode pode manipulá-las. Também não foi utilizada a operação de revogação, já que no SPKI/SDSI o uso de listas de certificados revogados (Certificate Revocation List- CRLs) não é recomendado.

está focada no modelo de confiança do X.509. A seguir, será descrito em detalhes esta integração, as funções de gerenciamento atribuídas aos serviços XKMS e o algoritmo de busca de certificado SPKI proposto neste trabalho.

4.1. Integração do serviço XKMS ao Modelo Federado do SPKI

No modelo de gerenciamento proposto, cada serviço XKMS serve apenas aos grupos de principais de sua federação SPKI, não participando ativamente de nenhuma cadeia de autorização. As chaves públicas de seus integrantes formam um grupo SDSI. A forma e a dinâmica de como o modelo de gerência suportado através das Federações SPKI é integrado ao serviço XKMS são analisadas a partir do exemplo ilustrado na Figura 2.

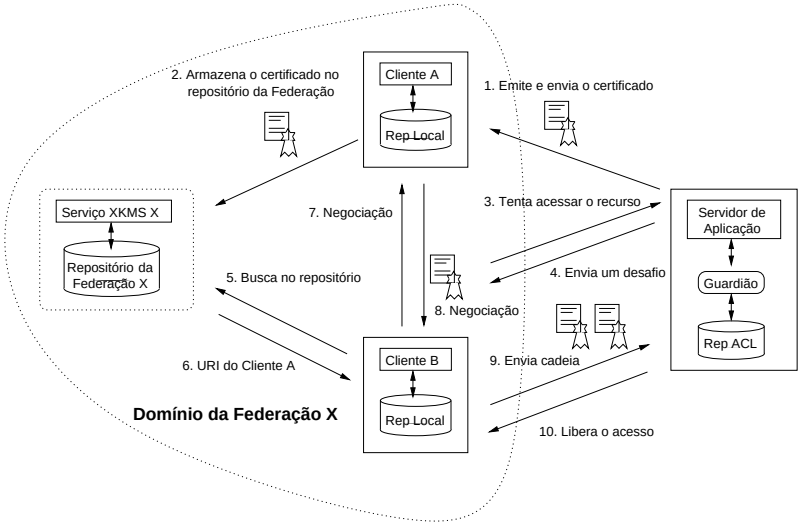

Figura 2. Integração do XKMS ao Modelo Federado do SPKI

No cenário da Figura 2, pode-se observar que os *clientes A e B* são membros da Federação X. No **passo 1**, o servidor de aplicação emite e envia um certificado de autorização delegável ao *cliente A*. Este cliente passa então a ter o direito de acesso ao recurso protegido pelo guardião, além de poder delegar este direito. No **passo 2**, o *cliente A* armazena este certificado no repositório da sua federação para que este esteja disponível aos demais membros. No **passo 3**, o *cliente B* tenta acessar o recurso do servidor de aplicação, porém o guardião do serviço verifica que este *cliente B* não possui o direito de acesso e responde com um desafio para que este prove que tem as permissões necessárias para acessar o recurso (**passo 4**), ou seja, que este apresente uma requisição assinada e uma cadeia de certificados que permita a checagem das autorizações concedidas. Diante do desafio recebido, o *cliente B* realiza, primeiramente, uma busca em seu repositório local. Como não encontra a cadeia de certificados, no **passo 5**, o *cliente B* envia uma requisição de localização para o seu serviço XKMS, para que este procure no repositório da sua federação. O serviço XKMS encontra o certificado que o *cliente A* armazenou no passo 2 e retorna a URI (*Uniform Resource Identifier*) do *cliente A* (**passo 6**). De posse da URI, o *cliente B* pode então negociar com o *cliente A* a delegação do direito de acesso (**passo 7**)[2]. Após a negociação, o *cliente A* emite um certificado de autorização

[2] A negociação de permissões pode ser realizada de maneira simples (através de uma delegação), já que os principais são membros de uma mesma federação SPKI. Entretanto, dependendo da aplicação, pode haver a necessidade de uma negociação mais complexa.

ao *cliente B* (**passo 8**) e, finalmente, o *cliente B*, de posse da cadeia de certificados que o liga ao serviço desejado, assina esta cadeia junto a uma requisição e envia ao servidor de aplicação (**passo 9**), que, por sua vez, libera o acesso ao recurso (**passo 10**).

4.2. Criação das Teias de Federações SPKI através dos Serviços XKMS Associados

Um principal, ao filiar-se a uma federação SPKI, pode fazer uso do repositório para requerer a localização de cadeias de certificados de autorização. Porém, uma federação SPKI tem sua abrangência limitada, havendo a necessidade de que os membros de uma federação se filiem a outras federações para alcançar certo nível de presença na rede. Para solucionar este problema, é proposto em [Santin 2004] que as federações, através de seus gerentes de certificados, se associem a outras federações, criando assim relacionamentos mútuos de confiança que permitem ampliar o modelo de confiança administrativo, formando as teias de federações SPKI. De forma semelhante, no modelo aqui proposto, uma federação SPKI pode se associar a outras federações, formando teias de federações SPKI, através dos seus Serviços XKMS.

Uma teia de federações SPKI é formada arbitrariamente[3] e cria caminhos de confiança entre as federações associadas de tal modo que um membro pode fazer consultas ao serviço XKMS de sua federação original e este serviço, por sua vez, realiza em nome de seu membro a busca nos Serviços XKMS de federações associadas.

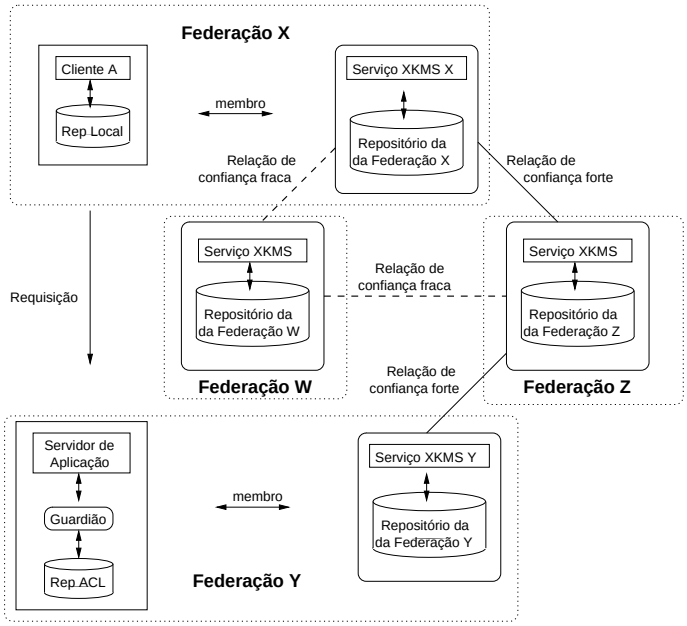

Figura 3. Criação das Teias de Federações através do XKMS

No cenário ilustrado na Figura 3, tem-se um *cliente A*, pertencente à *federação X*, que deseja acessar um servidor de aplicação pertencente à *federação Y*. O serviço XKMS da federação X é associado aos serviços XKMS das *federações W e Z*, e o serviço XKMS de *Z*, por sua vez é associado ao serviço XKMS da *federação Y*. Neste exemplo, o *cliente A* pode recorrer ao serviço XKMS da sua *federação X*, para que este realize a localização

[3]Uma vez que as mesmas são estabelecidas e removidas de forma dinâmica e aleatória.

através das teias de federações SPKI. Quando uma cadeia de certificados de autorização é localizada, o membro pode negociar a concessão do privilégio com o detentor do mesmo.

Um serviço XKMS é associado aos seus pares de outras federações com os quais tem uma relação de confiança. Esta relação de confiança pode ser de dois tipos: ligação *fraca* ou ligação *forte* de confiança. O tipo da ligação de confiança é definido através da regra de negócio. Por exemplo, uma ligação *forte* de confiança pode ocorrer entre uma federação de companhias de transportes aéreos e uma federação de hotéis. Estas federações podem apresentar fortes ligações de confiança por terem interesses comuns, como por exemplo, uma venda casada de passagens aéreas com estadias em hotéis. Caso as federações não apresentem esta relação forte de negócios, a ligação de confiança entre estas é considerada *fraca*.

No modelo de gerenciamento proposto, o serviço XKMS que se associa a uma outra federação SPKI, torna-se membro da mesma. Esta associação é concretizada pela emissão de um certificado de grupo SDSI de associados em cada federação envolvida. Ao serviço XKMS cabe, então, a manutenção das informações referentes aos membros e associados de sua Federação SPKI, removendo ou adicionando membros e associações com outras Federações SPKI, sem promover conflito de interesses. Estas funcionalidades do serviço XKMS são detalhadas na seção a seguir.

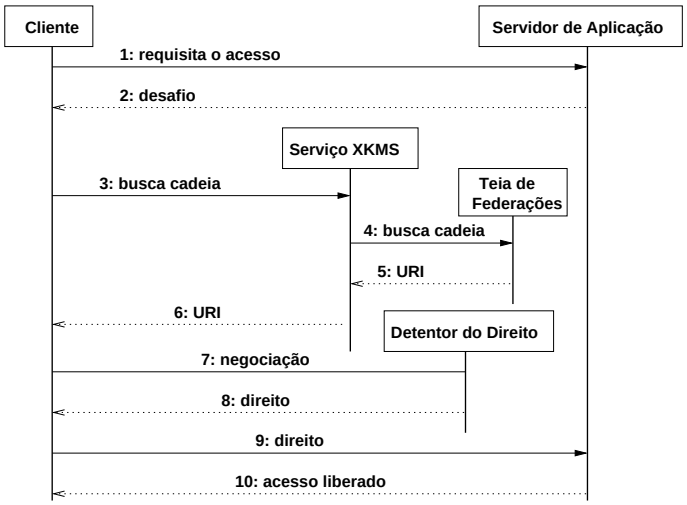

Figura 4. Troca de Mensagens para Formação de Novas Cadeias

As federações e suas teias formam o suporte para a geração de novas cadeias de certificados de autorização SPKI. O cenário da Figura 4 ilustra este processo de formação de novas cadeias, indicando as trocas de mensagens entre o cliente, o servidor de aplicação, serviços XKMS (teia de federações) e o detentor do privilégio. Os passos de 1 a 3, ilustrados na Figura 4, são semelhantes aos já discutidos na Figura 2. No passo 4, após verificar que a cadeia desejada não se encontra no repositório da federação, o Serviço XKMS, inicia a busca na teia de federações. O serviço XKMS associado, que encontrou a cadeia desejada, retorna a identificação do detentor do privilégio, uma URI (passo 5 e 6). De posse da identificação do detentor do direito, o próprio cliente negocia o direito desejado com o mesmo (passos 7, 8 e 9).

4.3. Operações Disponíveis no Serviço XKMS

XKRSS: Processos de Filiação, de Associação e de Registro de Certificados

Um principal pode filiar-se a quantas federações SPKI desejar. Para isto, conforme definido em [Santin 2004], o mesmo deve fornecer um endosso efetivado através da apresentação de um *threshold certificate* assinado por *k-dentre-n* membros definidos pela federação em questão [4]. A cada novo membro da Federação SPKI é emitido um novo certificado de grupo, expressando a participação na Federação, para fins de comprovação da filiação (*membership*). Do mesmo modo, um serviço XKMS se torna um associado de outra federação ao fornecer o endosso assinado por *k-dentre-n* associados.

No modelo de gerenciamento federado, o processo de filiação se dá através da operação de registro do XKRSS. Conforme ilustrado na Figura 5, esta mesma operação é utilizada para a filiação de membros, para a submissão de certificados delegáveis no repositório da federação SPKI e para estabelecer associações entre federações. Desta forma, quando um serviço XKMS recebe uma requisição de registro, o mesmo verifica se no conteúdo da mensagem encontra-se um *threshold certificate* ou uma cadeia de certificados de autorização para então identificar qual tarefa deverá realizar.

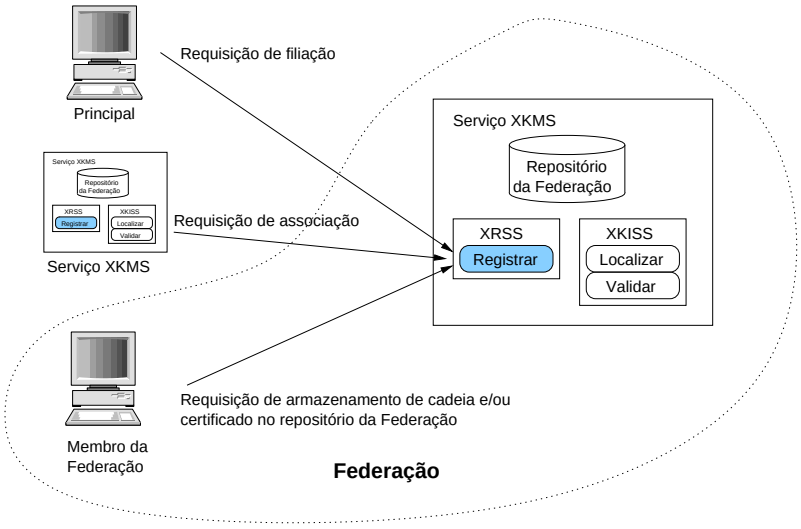

Figura 5. Operações do Serviço XKMS

XKISS: Localização e Validação de Certificados nas Teias de Federações

A operação de localização de um serviço XKMS é utilizada pelos membros da federação e pelos serviços XKMS associados para requisitar uma busca de cadeias de certificados de autorização na sua federação. Um membro também pode requisitar a um serviço XKMS que valide uma cadeia de certificados, por exemplo, de autorização, utilizando a operação *validar*.

O uso dos serviços XKMS facilita a localização de certificados de autorização através de buscas realizadas nas teias de federações SPKI. A heurística para a navegação nas teias de federações é definida e detalhadamente descrita na seção, a seguir.

[4] Esta estratégia facilita o processo de filiação no sentido de permitir que um subconjunto k de principais contido no conjunto n possa atuar no papel de administrador da federação.

4.4. Algoritmo de Busca de Certificados nas Teias de Federações

A fim de ilustrar a dinâmica do algoritmo proposto de busca de cadeias de autorização, é apresentado um cenário na Figura 6, no qual um *membro* da *Federação A* solicita ao seu serviço XKMS a localização de uma cadeia de autorização. Após constatar que a cadeia não se encontra no repositório da sua federação, este serviço inicia o algoritmo de busca de certificados na teia de federações ilustrada, através do seus serviços XKMS associados.

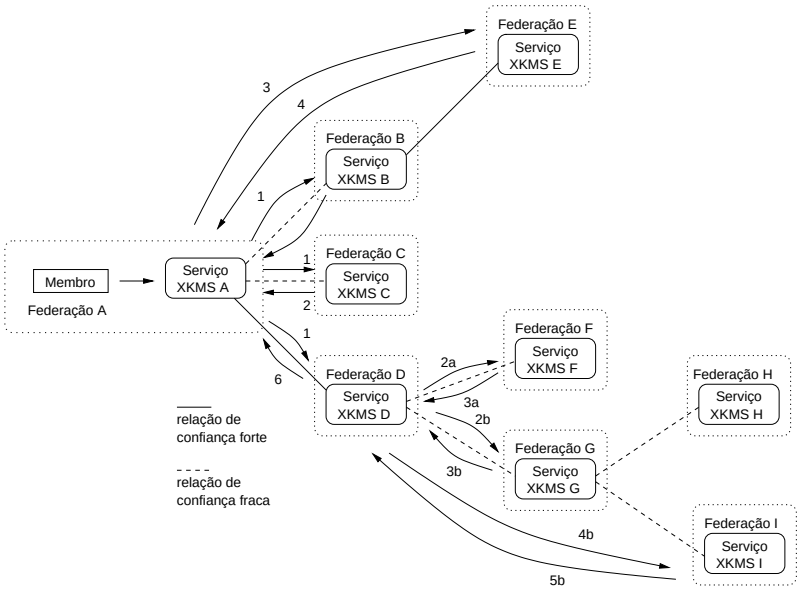

Figura 6. Um Cenário de Uso do Algoritmo de Busca de Certificados nas Teias de Federações

Conforme descrito anteriormente, no modelo proposto, uma ligação de confiança entre serviços XKMS pode ser de dois tipos: *fraca* ou *forte*. Quando a relação de confiança é ***fraca***, o funcionamento do algoritmo de busca de certificados é realizado de forma interativa. Isto é, quando o serviço *XKMS A* envia uma requisição aos seus serviços XKMS associados, cuja relação de confiança é fraca (serviços *XKMS B* e *XKMS C* - **passo 1**, Figura 6). Caso esses serviços constatem que a cadeia de autorização desejada não está registrada em sua federação, estes retornam as URIs dos seus serviços XKMS associados para que o próprio serviço *XKMS A* continue a busca na teia de federações (**passo 2**).

Se a relação de confiança entre os serviços associados for ***forte*** (como as relações entre o *XKMS A* e *XKMS D*), a busca é feita de forma recursiva. Neste caso, o serviço *XKMS A* repassa a missão de localizar a cadeia para o seu associado forte (**passo1**). Ou seja, o serviço *XKMS D* verifica se a requisição veio de um serviço com quem este tenha uma relação de confiança forte e, em caso afirmativo, este assume a busca da cadeia em seus associados (**passos 2a e 2b**). Desta forma, a busca é realizada como se um dos seus membros a tivesse requisitado. No **passo 3b**, o serviço *XKMS G* associado ao serviço *XKMS D*, também não encontra a cadeia e repassa as URIs dos serviços *XKMS H* e *XKMS I* associados ao mesmo. Através destas URIs, **passo 4b**, o serviço *XKMS D* envia uma requisição de localização de certificados ao *XKMS I*, que finalmente retorna uma mensagem indicando que a cadeia foi encontrada e informa a URI do membro da sua federação que detém o recurso (cadeia desejada). Na localização recursiva, o serviço

XKMS, ao delegar a tarefa de busca de cadeias de certificados a um serviço associado, diminui a sua carga de processamento para concretizar a operação de localização.

Ao visualizar a teia de federações SPKI como um grafo, tem-se que a busca de certificados pode ser de duas formas: a busca por amplitude e a busca por profundidade. No modelo proposto, optou-se pela busca em amplitude, pois quanto mais distante o requerente do principal detentor do direito estiver, maior a dificuldade de negociação, pois estes pertencem a federações distantes; por conseqüência, a probabilidade de os membros possuírem interesses afins é menor. Para que a busca não seja infinita, foi inserido também um controle de saltos que é passado na requisição de busca do certificado (*ttl*).

O protocolo proposto, que contempla a dinâmica apresentada possui duas mensagens: *locate*, usada para efetuar a localização da cadeia de certificados; e a *locateHit*, que informa se a cadeia procurada foi encontrada. A mensagem *locate* é composta por três elementos principais: o recurso, que identifica a cadeia de certificado que está sendo requisitada; a assinatura da mensagem, onde se obtém a chave pública para identificação do requerente e o *ttl*, usado para controlar o número de saltos possíveis executados na busca. A seguir, o algoritmo para localização de cadeias de autorização através das teias de federações SPKI (trocas entre XKMSs) é apresentado.

Algoritmo 1 locate$(recurso, sign, ttl)$

Require: $T = \{$ Conjuto de serviços com associação forte. $\}$
Require: $U = \{$ Conjuto de serviços com associação fraca. $\}$
1: **if** $(ttl > 0) AND (signature.getPublicKey() \in (T \cup U))$ **then** {A requisição deve ser proveniente de um serviço com associação forte ou fraca e ttl tem que ser maior do que zero.}
2: ttl ← ttl-1
3: uris ← buscaRepositorio(recurso)
4: **if** $(uris \neq \varnothing)$ **then** {Algum membro possui o recurso}
5: locateHit(uris, "members")
6: **else if** $(signature.getPublicKey() \in U)$ **then** {A requisição é de um serviço de associação fraca}
7: uris = associatedURIs($T \cup U$){URIs dos associados}
8: locateHit(uris, "associateds")
9: **else if** $(ttl > 0)$ **then** {Busca nos associados com relação forte e fraca}
10: $N \leftarrow T \cup U$
11: **while** $N \neq \varnothing$ **do**
12: x ← firstElement(N)
13: x.locate(recurso, sign(), ttl)
14: $N \leftarrow N \setminus \{x\}$ {Remove o elemento x do conjunto N}
15: **end while**
16: **end if**
17: **else**
18: locateHit($\varnothing$,"")
19: **end if**

5. Implementação e Resultados

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 e, por conseguinte, validar o modelo proposto. Para compor a camada necessária para o desenvolvimento de aplicações baseadas na arquitetura dos Serviços *Web*, escolheu-se o *framework* de código aberto *Apache Axis* e o Tomcat[5] como servidor de aplicação. Para compor a camada de qualidade de proteção e prover segurança as mensagens trocadas entre cliente e serviço XKMS e entre serviços XKMS, adotou-se as implementações abertas das

[5]http://ws.apache.org/axis e http://tomcat.apache.org, respectivamente.

especificações *WS-Security* , *XMLEnc*, *XMLDSign*, as bibliotecas *Apache WSS4J*[6] e *Apache XML Security*[7].

A especificação XML Signature [Bartel et al. 2002] prevê que o elemento `<ds:KeyInfo>` contenha um subelemento chamado `<ds:SPKIData>`, que deve ser usado para transmitir pares de chaves, assim como certificados ou outros dados associados ao SPKI. Entretanto, esta especificação não define como estes dados SPKI são inseridos neste elemento. Diante disto, propõe-se neste trabalho uma extensão do *XML Signature Schema* de modo a definir quais elementos da infra-estrutura SPKI/SDSI podem ser inseridos em uma assinatura XML[8]. O elemento `<ds:SPKIData>` contém o elemento `<ds:SPKISexp>` que passa então a conter os subelementos, definidos na Tabela 1, conforme especificado no código XML a seguir.

```
1  <element name="SPKIData" type="ds:SPKIDataType"/>
2  <complexType name="SPKIDataType">
3    <sequence maxOccurs="unbounded">
4     <element name="SPKISexp" type="ds:SPKISexpType"/>
5      <any namespace="##other" processContents="lax" minOccurs="0"/>
6    </sequence>
7  </complexType>
8  <element name="SPKISexp" type="ds:SPKISexpType"/>
9  <complexType name="SPKISexpType">
10   <sequence maxOccurs="unbounded">
11     <choice>
12       <element name="SPKIKeyValue" type="ds:KeyValueType"/>
13       <element name="SPKITag" type="spki:tag-content"/>
14       <element name="SPKIUris" type="spki:uris"/>
15       <element name="SPKIAuthorization-cert" type="
            spki:authorization-cert"/>
16       <element name="SPKIName-cert" type="spki:name-cert"/>
17       <element name="SPKISequence" type="spki:sequence"/>
18     </choice>
19   </sequence>
20 </complexType>
```

Figura 7. Elemento *SPKIData* do *XML Signature Schema*

A infra-estrutura SPKI/SDSI, integrante da arquitetura do protótipo, é responsável pela criação, emissão e gerenciamento dos certificados SPKI/SDSI, pela verificação das cadeias de autorização e por oferecer o suporte necessário para as Federações SPKI. Uma biblioteca de códigos Java que implementa o SPKI/SDSI, chamada JSDSI2.0 [Morcos 1998], foi utilizada. De forma a prover o suporte as Federações SPKI e tornar esta infra-estrutura mais flexível e eficiente para ser usada pelo XKMS, a biblioteca JSDSI foi estendida dentro do contexto do projeto Cadeias de Confiança[9].

[6]http://ws.apache.org/wss4j/

[7]http://xml.apache.org/security

[8]Estas extensões foram implementadas na biblioteca *XML Security*, seguindo as recomendações para extensões definidas pela *Apache Foundation*. Assim que a fase de testes de software for concluída, esta extensão, bem como a extensão na biblioteca WSS4J, que tornam possível o uso da infra-estrutura SPKI/SDSI nestas bibliotecas, serão submetidas a *Apache*.

[9]CNPq/PROTEM- Conteúdos Digitais - http://www.das.ufsc.br/seguranca

`KeyValue`	Valor de uma chave pública
`tag`	Expressão real que pode transmitir autorizações
`uris`	Lista de URIs
`authorization-cert`	Certificado de autorização SPKI.
`name-cert`	Certificado de nome SPKI
`sequence`	Cadeia de autorização SPKI

Tabela 1. Subelementos do `<ds:SPKISexp>`

A especificação do XKMS define diferentes modos de troca de mensagens entre os serviços XKMS e seus clientes. Neste trabalho, para todas as trocas de mensagens entre os serviços XKMS e entre um membro e um serviço, foi implementado o protocolo assíncrono de duas fases. Todas as operações do XKISS e XKRSS, definidas no modelo proposto, incluindo o algoritmo de busca na teia de federações, foram implementados. Visando a facilitar o uso, o entendimento e o gerenciamento da infra-estrutura SPKI, incluindo as operações para gerenciamento das Federações SPKI através do XKMS, a interface gráfica proposta em [Morcos 1998] foi estendida[10].

6. Trabalhos Relacionados

A maioria dos trabalhos descritos na literatura, que fazem uso de um serviço XKMS, utilizam somente a ICP X.509, herdando com isso os problemas de uma abordagem de autenticação centralizada. Entre estes trabalhos, destaca-se o serviço de validação de certificados para *grids* computacionais apresentado em [Park et al. 2003]. Este serviço introduz um módulo de validação de certificados (CVM), usando o serviço XKMS, no qual vários protocolos são utilizados, tais como: o OSCP (*Online Status Certificate Protocol*), o SCVP (*Standard Certificate Validation Protocol*) e o LDAP (*Lightweight Directory Access Protocol*). Nos testes de validação de certificados, devem ocorrer também as verificações das CRL's. Outro trabalho relevante descrito em [Bilykh et al. 2003] apresenta um *grid* de informações de saúde chamado *HealthInfo Grid* que possui um componente chamado *Medical Exchange Agency* (MEA). Este componente, usa ICP X.509 e assume o papel de autoridade certificadora (AC) para os demais componentes do *grid* e para cada um destes componentes, o MEA emite um certificado de rede. Todo o o gerenciamento destes certificados é feito através da interface XKMS. Outros trabalhos que fazem uso do XKMS, combinado a ICP X.509, para prover facilidades para aplicações baseadas em serviços são [Kraft 2002, Adams e Boeyen 2002, Daniel J. Polivy 2002]

Os trabalhos citados acima comprovam a relevância em se utilizar o serviço XKMS para abstrair a manipulação das ICP das aplicações distribuídas. Entretanto, em todos estes trabalhos somente a ICP X.509 é considerada. Todas as implementações do XKMS, independentes de uma aplicação distribuída, como por exemplo o *SQLData XKMSServer*[11], *Markup Security Project*[12], *XKMS Prototype Server*[13] e a implementação da Verisign[14] não oferecem suporte para o SPKI/SDSI. O protótipo descrito neste trabalho

[10]Mais informações sobre a implementação do protótipo, bem como o seu código fonte, estão disponíveis no endereço *www.das.ufsc.br/seguranca/servicoXKMS*.

[11]http://www.sqldata.com/xkms.htm

[12]http://markupsecurity.com/info/xkms/index.htm

[13]http://www.wingso fhermes.org/xkms.html

[14]http://www.xmltrustcenter.org/xkms/index.htm

é a primeira implementação do XKMS para manipular certificados SPKI/SDSI.

A infra-estrutura SPKI/SDSI necessita de um modelo de gerência para auxiliar os principais na localização de uma cadeia de autorização necessária para o acesso a determinado recurso. Em [Santin 2004], foi descrito um modelo de gerência, porém neste modelo há uma sobrecarga no cliente que, além de ter que entender a complexidade da ICP, também é o responsável pela tarefa de localização de cadeias de certificados. Além disso, o acesso às funcionalidades providas pelo mesmo não segue um modelo de mensagens padrão. A proposta descrita neste artigo engloba o modelo de gerência proposto em [Santin 2004], porém o serviço XKMS, em substituição ao gerente de certificados, absorve a responsabilidade da localização dinâmica dos caminhos de confiança. A solução para o uso e gerenciamento federado do SPKI, através de um Serviço *Web*, provê uma padronização para o acesso as funcionalidades oferecidas em uma Federação SPKI.

7. Conclusão

A habilidade de definir grupos e de delegar autorizações, bem como as facilidades para o desenvolvimento de sistemas distribuídos escaláveis e seguros, fazem do SPKI/SDSI uma boa opção para o desenvolvimento de aplicações distribuídas baseadas em redes de confiança. Neste trabalho, foi apresentada uma abordagem para integrar o modelo de confiança do SPKI/SDSI aos padrões de serviços *Web* e de extensões XML. A partir do conceito de Federações SPKI/SDSI, o modelo proposto oferece um gerenciamento federado de certificados SPKI, através da tecnologia orientada a serviços, via XKMS. A localização de certificados atribuída ao serviço XKMS poupa o cliente desta custosa tarefa. Este modelo, por oferecer um Serviço *Web* XKMS, apresenta um protocolo padrão para o acesso às funções de gerenciamento e estabelecimento de confiança.

Na literatura, não existe nenhum esforço do uso do XKMS com o SPKI. Neste ponto, o modelo proposto neste artigo é original. A explicação para a ausência de abordagens concorrentes é que a especificação do XKMS embora, possua elementos nos quais é possível o uso de diferentes ICPs, na prática, esta foi toda definida visando uma ICP hierárquica como o X.509. A integração do XKMS ao modelo de confiança do SPKI não foi uma tarefa simples, porém não houve nenhuma alteração ou extensão que comprometesse a conformidade com a especificação XKMS [Hallam-Baker e Mysore 2005].

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 agracem aos membros deste projeto por suas contribuições e ao CNPq pelo suporte financeiro.

Referências

Adams, C. e Boeyen, S. (2002). Uddi and wsdl extensions for web service: a security framework. In *Proceedings of the 2002 ACM workshop on XML security*, pages 80–89.

Bartel, M., Boyer, J., e Fox, B. (2002). *XML-Signature Syntax and Processing*. W3C. `http://www.w3.org/TR/xmldsig-core`.

Bilykh, I., Bychkov, Y., Dahlem, D., Jahnke, J. H., McCallum, G., Onabajo, C. O. A., e Kuziemsky, C. (2003). Can grid services provide answers to the challenges of national health information sharing? In *Proceedings of the 2003 conference of the Centre for Advanced Studies on Collaborative research*, pages 01–15.

Clarke, D. E. (2001). Spki/sdsi http server/certificate chain discovery in spki/sdsi. Master's thesis, Massachusetts Institute of Technology - MIT.

Daniel J. Polivy, R. T. (2002). Authenticating distributed data using web services and xml signatures. In *Proceedings of the 2002 ACM workshop on XML security*, pages 80–89.

Ellison, C. M., Frantz, B., Lampson, B., Rivest, R., Thomas, B. M., e Ylonen, T. (1999). *SPKI Certificate Theory*. IETF RFC 2693.

Hallam-Baker, P. e Mysore, S. H. (2005). *XML Key Management Specification (XKMS 2.0)*. W3C – Proposed Recommendation.

Imamura, T., Dillaway, B., e Simon, E. (2002). *XML Encryption Syntax and Processing*. W3C. `http://www.w3.org/TR/xmlenc-core`.

Kraft, R. (2002). Designing a distributed access control processor for network services on the web. *ACM Transactions on Information and System Security (TISSEC)*, 7(1):36–52.

Morcos, A. (1998). A java implementation of simple distributed security infrastructure. Master's thesis, Massachusetts Institute of Technology.

OASIS (2005). *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*. Organization for the Advancement of Structured Information Standards (OASIS). `docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf`.

Park, N., Moon, K., e Sohn, S. (2003). Xml security: Certificate validation service using xkms for computational grid. In *Proceedings of the 2003 ACM workshop on XML security*, pages 112–120.

Rivest, R. L. e Lampson, B. (1996). SDSI – A simple distributed security infrastructure. Presented at CRYPTO'96 Rumpsession.

Santin, A. (2004). *Teias de Federações: uma Abordagem baseada em Cadeias de Confiança para Autenticação, Autorização e Navegação em Sistemas de Larga Escala*. PhD thesis, Universidade Federal de Santa Catarina.