INTEGRANDO O MODELO DE SEGURANÇA *SPKI/SDSI* AO AMBIENTE DE GERÊNCIA *WBEM*

Darlan Paulo Siqueira Carrião, Altair Olivo Santin, Carlos Alberto Maziero

Programa de Pós-Graduação em Informática Aplicada – Centro de Ciências Exatas e de Tecnologia – Pontifícia Universidade Católica do Paraná (PUC-PR)
Rua Imaculada Conceição, 1155 – 80.215-901 – Curitiba – PR – Brasil
{darlan, santin, maziero}@ppgia.pucpr.br

Resumo

Resumo. *Este trabalho apresenta a integração do modelo de autorização e autenticação SDSI/SPKI ao padrão de gerenciamento WBEM proposto pelo Distributed Management Task Force (DMTF). Através da integração proposta é oferecida maior portabilidade e facilidade na administração de políticas de autorização. As cadeias de certificados SPKI dispensam o uso de autoridade certificadora centralizando a autenticação. Busca-se mostrar que a integração proposta eliminaria a necessidade da relação de confiança inter-domínios, garantindo desta maneira a interoperabilidade entre os diversos servidores WBEM. O protótipo implementado mostrou a efetividade da proposta em acessos a objetos CIM gerenciados pela Internet nos teste realizados.*

Abstract. *This work presents the integration of the SDSI/SPKI authorization and authentication model to the standard of management WBEM considered for Distributed Management Task Force (DMTF). Through the integration proposal is offered to bigger portability and easiness in the administration of authorization policies. The chains of SPKI certifyd excuse to the authority use certifier centering the authentication. One searchs to show that the integration proposal would eliminate the necessity of the reliable inter-domains relation, guaranteeing in this way the interoperability between diverse WBEM servers. The implemented archetype showed to the effectiveness of the proposal in accesses objects CIM managed by the Internet in the test carried through.*

1. Introdução

A preocupação crescente com a qualidade dos serviços oferecidos pelos sistemas de informação tem exigido uma melhor gestão dos recursos informáticos. Isto implica em monitorar e administrar um leque cada vez maior e mais complexo de usuários e recursos de software, hardware e de infra-estrutura que tomam a forma de uma grande base de informações.

Atualmente, estão em uso vários sistemas padronizados de gerenciamento destas informações, porém na integração entre os mesmos, dentro de um ambiente distribuído e heterogêneo, várias dificuldades de interoperabilidades podem acontecer.

O modelo comum de informação (*CIM*), com base no paradigma da orientação a objetos, define uma abstração do ambientes de gerenciamento que abrange todos os seus níveis, suportando também a integração dos sistemas padronizados atualmente em uso.

A segurança da informação se torna uma tarefa mais difícil à medida que o *CIM* define gerenciamento corporativo (*enterprise*) baseado na *Web* (*WBEM*). O fluxo de informações trafegando sobre *HTTP* demanda esquemas de autenticação e autorização mais adaptados a Internet. O sistema de autenticação clássico depende de um servidor de nomes que representa

um domínio e só consegue escalabilidade através da construção de relações de confiança interdomínio. No caso de *PKIs*, como a *X.509*, a confiança é herdada quando há compartilhamento da mesma autoridade certificadora raiz (*root*), em caso contrário, também há dependência de estabelecimento de relações de confiança interdomínio. Este esquema obriga o serviço de autorização a compartilhar as mesmas entidades de confiança do serviço de autenticação. Segurança baseada na dependência de várias relações de confiança pode criar pontos de vulnerabilidade no esquema.

Um esquema de segurança com escalabilidade facilitada, independente de uma entidade de confiança, mais flexível e com portabilidade de política de autorização se faz necessário.

Nesta proposta será mostrado um esquema de autenticação e autorização e, definição de políticas de autorização que utiliza chaves públicas na identificação de principais. A assinatura digital é utilizada como mecanismo de autenticação. Com um modelo de confiança baseado na construção de cadeias de certificados de autorização, o *SPKI / SDSI* rompe com a necessidade do serviço de nomes e da definição de domínios de segurança e, conseqüentemente se mostra mais adequado ao ambiente *Web*. As cadeias de autorização são construídas através da delegação de direitos de um principal para outro e assim as políticas de autorização ficam distribuídas pela rede e não concentradas num servidor. Este esquema integra-se ao ambiente de gerenciamento baseado na *Web* (*WBEM*), como uma alternativa bastante vantajosa se comparado aos sistemas clássicos de segurança utilizados atualmente.

Este trabalho está estruturado da seguinte forma: na seção 2 será abordado o modelo de informação comum (*CIM*), na seção 3 será apresentado o *WBEM*, na seção 4 será abordado o *SPKI / SDSI* e na seção 5 será discutida a proposta e implementação deste trabalho terminando na seção 6 com os trabalhos correlatos.

2. Modelo Comum de Informação (*CIM*)

O modelo comum de informação [DMTF, 1999] é uma visão conceitual de um ambiente de gerenciamento, que tenta unificar e estender padrões já existentes de gerenciamento como, por exemplo: *Simple Network Management protocol* (SNMP) [Stallings, 1994], *Desktop Management Interface* (DMI) [DMTF, 2003a] e *Common Management Information Protocol* (CMIP) [Stallings, 1994]. Estas extensões, padronizadas pelo comitê *Distributed Management Task Force* (DMTF), se fazem principalmente no sentido da representação das informações segundo o paradigma da orientação a objetos. O objetivo principal desta padronização é facilitar a interoperabilidade e o compartilhamento de informações entre os diversos ambientes de gerenciamento.

A coleta, o armazenamento e o compartilhamento de informações de gerenciamento dentro de um ambiente de natureza heterogênea e distribuída, exigem um modelo de segurança flexível, distribuído, simples e com interoperabilidade facilitada. Na seção 2.1 será abordado brevemente o suporte de segurança, na seção 2.2 serão considerados aspectos de interoperabilidade e na seção 2.3 será apresentado o esquema de nomeação de objetos que o modelo *CIM* oferece para um ambiente de gerenciamento com as características citadas acima.

2.1 Suporte de segurança

As principais classes que representam a segurança do modelo *CIM*, são encontradas nos modelos: de usuário e segurança, política e, de interoperabilidade. Todas as classes *CIM* são heranças de uma meta classe chamada *ManagedElement*. Todos os modelos são estendidos do modelo essencial e do modelo comum do *CIM*.

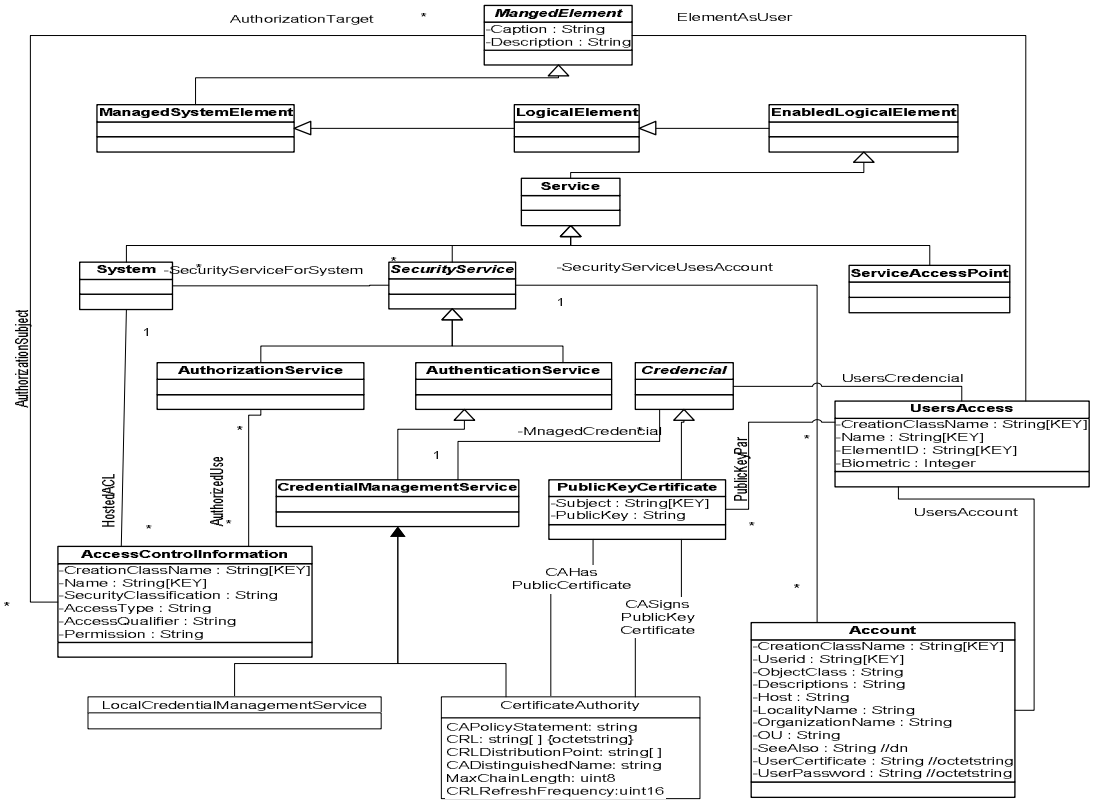

Figura 1 - Autorização e autenticação CIM

O modelo *CIM* da Figura 1 representa em um único diagrama *UML*(*Unified Modelling Language*)[OMG, 1999] as classes diretamente ligadas a autenticação e autorização[DMTF, 2000b].

Na verdade a classe *SecurityService* da Figura 1 é a classe base para segurança no modelo *CIM*. Desta classe derivam as classes para os serviços de autenticação (*SecuritySevice*) e autorização (*AuthorizationService*). O serviço de autenticação se baseia nas informações do usuário (nome e identificação do usuário – *Name* e *Userid*, respectivamente) fornecidas através de credenciais (*Credentials*) por *UsersAccess* e *Account*. O nome também pode estar ligado a uma chave pública (*PublicKey*) codificada em um certificado (*PublicKeyCertificate*). Neste caso, a identificação do usuário pode ser provida por uma autoridade certificadora (*CA*) através de um certificado assinado digitalmente por tal *CA*. As informações de autorização para controle de acesso são fornecidas pela classe *AccessControlInformation* na implementação das respectivas *ACLs* (listas de controle de acesso).

2.2 Modelo de Interoperabilidade

O modelo de interoperabilidade *CIM* [DMTF, 2003b] provê um modelo de informação para descrever negócios e informática em ambientes organizacionais e de Internet. O modelo fornece definições e estruturas consistentes de dados, utilizando técnicas orientadas a objetos, para descrever as características comuns de gerenciamento e dos componentes de um servidor *WBEM* (*Web Based Enterprise Management*). O modelo reflete as classes e propriedades do servidor *WBEM* que são independentes de implementação.

O modelo de interoperabilidade, que descreve a arquitetura lógica do servidor *WBEM*, está dividido em quatro sub-modelos. O sub-modelo para *namespace* (que será abordado na seção 2.3). O sub-modelo do gerenciador de objetos *CIM* (*CIMOM*) e suas capacidades (que será detalhado na seção 3.2). Alem destes, há ainda o sub-modelo do adaptador de protocolos, que

descreve as diferentes adaptações dos protocolos aos serviços e faz as respectivas associações aos mecanismos de comunicação e, o sub-modelo para as estatísticas do servidor que define estatísticas baseadas nas operações *WBEM*.

2.3 Espaços de nomes de objetos (*Namespaces*)

Um *namespace* é uma abstração lógica de classes, associações e instâncias que delimitam a visibilidade e o escopo de um objeto para se adaptar a um ambiente gerenciado particular. Para obter tal delimitação, pode-se agrupar um conjunto de classes de um mesmo proprietário, que então passarão a ser referenciadas como um esquema. Os esquemas são utilizados para administração e nomeação de classes. Os nomes de classes devem ser únicos dentro do esquema a que pertencem.

Um esquema *CIM* é um modelo de objetos baseado em propriedades chave de uma classe – expressas através do qualificador *KEY*. Todas as instâncias de classes são nomeadas e referenciadas unicamente pelas chaves da classe. Associações de classes são unicamente identificadas por suas chaves, as quais sempre incluem suas propriedades de referência. A referência consiste de chaves de classes e valores das instâncias para estas chaves. Todas as classes concretas devem definir ou herdar uma estrutura de chaves. Se a estrutura for herdada a mesma não pode ser alterada.

Figura 2 - Nomeação de objeto CIM

Um *namespace* pode conter diferentes classes de diferentes esquemas. O nome de um objeto *CIM* consiste do caminho do *namespace* e de um modelo de caminho (como mostra a Figura 2). Para permitir a navegação completa dentro dos esquemas *CIM*, o caminho do *namespace* é utilizado para localizar a implementação de um objeto. O modelo de caminho permite localizar o objeto dentro do *namespace* e é dependente de implementação.

Se os objetos forem implementados num serviço de diretório como o *X.500* [Wahl, 1997], por exemplo, através de sua implementação para a pilha de protocolo *TCP/IP*, o *LDAP* [Howes, Wahl e Kille, 1997]. Então, o modelo de caminho será baseado em *Distinct Name* (DN). Uma outra alternativa de modelo de nome é o uso do qualificador *KEY* disponível em todas as classes onde a identificação deve ser única (Exemplo da Figura 2).

3. *Web Based Enterprise Management* (*WBEM*)

O padrão de gerenciamento baseado na *WEB* (*WBEM*) é um conjunto de tecnologias padronizadas, desenvolvido pela *DMTF*, para integrar tal prática. Através de interfaces comuns para todos os sistemas, o *WBEM* simplifica e reduz custos sem precisar substituir os sistemas atuais de gerenciamento, pois o *WBEM* se integra facilmente com os mesmos.

3.1 Componentes *WBEM*

A arquitetura lógica do *WBEM* é composta por vários componentes como pode ser visto na Figura 3. O Cliente *WBEM* requisita acesso aos objetos gerenciados, através de uma operação *XML* [DMTF, 1999] – traduzida pelo adaptador de protocolo de *XML* para *CIM* e entregue ao Gerenciador de Objetos *CIM* (*CIMOM*).

O *CIMOM* é o componente central do servidor *WBEM*, responsável por receber e processar as solicitações de operações feitas pelo cliente e pela comunicação entre todos os componentes do servidor. O manipulador de indicação *WBEM* faz o roteamento e a entrega de mensagens do *CIMOM* para os respectivos destinos. Os provedores *WBEM* são agentes de gerenciamento externos (*SNMP, CMIP,...*) ou interno (*CIM*) que monitoram elementos gerenciados como um *router*, por exemplo.

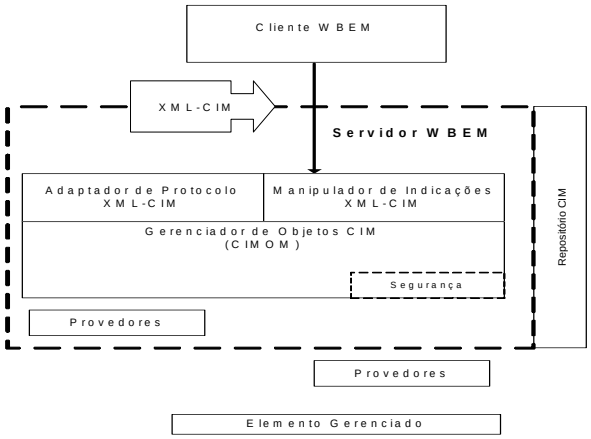

Figura 3 - Arquitetura lógica do servidor WBEM

3.2 *CIMOM*

O *CIMOM* tem um repositório onde estão armazenados todos os esquemas *CIM* que servem para verificação dos dados enviados pelo cliente e pelo provedor. Este repositório pode ser utilizado também para criação de instância de dados *CIM* a partir de clientes ou de provedores. As instâncias de dados salvas desta maneira são chamadas de dados estáticos.

Os dados coletados pelo provedor a partir de elementos gerenciados são denominados dados dinâmicos. Assim, quando um cliente precisa modificar ou acessar dados estáticos o *CIMOM* trabalha com o seu repositório. Porém, quando a operação for em dados dinâmicos o *CIMOM* faz o devido encaminhamento ao provedor envolvido com a requisição.

A segurança efetivada no *CIMOM* é dependente de implementação – não é especificada pela *DTMF* – embora na definição do *CIMOM* exista a classe *ServerSecurity* e a interface *CommonServerSecurityContext* que podem ser utilizadas como ferramentas para manipular a autenticação e autorização no *WBEM.*

4. *SPKI* (*Simple Public Key Infrastructure*) / *SDSI* (*Simple Distributed Security Infrastructure*)

O desenvolvimento do *SDSI* (*Simple Distributed Security Infrastructure*) e do *SPKI* (*Simple Public Key Infrastructure*) foi motivado pela limitação e pela complexidade da infra-estrutura de chaves publicas baseada na hierarquia global de nomes, *X.509.* O *SPKI* [Ellison e outros, 1999] é uma infra-estrutura de chaves públicas que tem como características o uso de *namespaces* locais. O *SPKI* [Lampson and Rivest, 1996] foi desenhado com a intenção de ser um modelo de autorização simples e flexível, muito bem definido e de fácil implementação. A união do *SPKI* e *SDSI* resultou em um sistema de autenticação e autorização para aplicações distribuídas. Com a criação desta infra-estrutura os espaços de nomes de principais[1] são locais e o modelo, baseado em cadeias de confiança, é simples e flexível.

[1] Entidades ativas que possuem um par de chaves (privada e pública) e através disto podem executar assinaturas digitais.

Em *SPKI / SDSI* existem dois tipos distintos de certificados: para nomes e para autorizações. Os certificados de nomes são responsáveis por associar nomes a chaves públicas ou a outros nomes. O sistema de nomeação é adotado do *SDSI* que induz ao uso de nomes locais mesmo no sentido global de um ambiente distribuído. Os nomes *SPKI / SDSI* são sempre locais, correspondendo ao espaço de nomes de quem emitiu o certificado. O emissor do certificado é sempre identificado pela sua chave pública. A combinação chave pública mais nome local forma um identificador global único.

No *SPKI / SDSI* é usado um modelo igualitário: os principais são chaves públicas que podem assinar e divulgar certificados, como uma *CA* do *X.509*. Assim, qualquer principal pode criar seu par de chaves (privada e pública), e então, associar à chave pública do par a um nome no seu espaço local de nomes e divulgá-los através de certificados, o que exclui a necessidade de uma entidade centralizadora que faça o registro de chaves públicas e emita certificados como a *CA* da *PKI X.509*. Assim, cada principal define da maneira que lhe parecer mais intuitiva, em seu espaço de nomes, os nomes atribuídos a um outro principal.

Um certificado de nomes pode fazer referência a um nome publicado num certificado, no espaço de nomes de outro principal e assim sucessivamente, de modo a formar uma cadeia de certificados de nome. Assim, a divulgação de nomes no *SPKI / SDSI* é feita através de redes de confiança formadas por certificados de nomes ligados por encadeamento de referências(*linked names*). Estas cadeias de nomes devem ser reduzidas a uma chave pública que representa o principal sendo referenciado quando se deseja a identificação do mesmo.

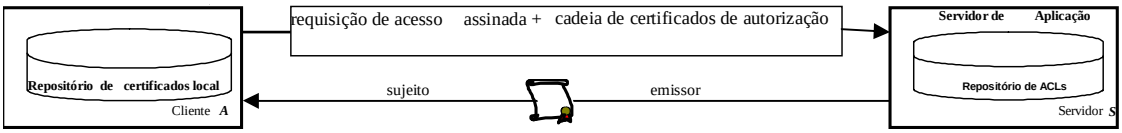

Figura 4 - Fluxo de autorização SPKI

Os certificados de autorização *SPKI / SDSI* ligam autorizações a um nome ou a uma chave. Através destes certificados, o emissor delega permissões de acesso a outros principais no sistema. Na infra-estrutura *SPKI / SDSI* os certificados de autorização são construídos a partir das *ACL's* do guardião (monitor de referência). O conteúdo do certificado pode ser o mesmo da *ACL*, porém, ao certificado é acrescido o campo do emissor assinando o certificado – a *ACL* não possui este campo porque é local ao guardião do serviço.

A Figura 4 mostra o fluxo de autorização baseado no modelo confiança *SPKI / SDSI*. Através da delegação de privilégios, a partir do servidor de aplicação, é criado um caminho de autorização que forma a rede de confiança entre o servidor e o cliente. Na Figura 4, o cliente *A*, após receberem os certificados indicados, terá a cadeia de autorização necessária para o acesso ao servidor. A cadeia de autorização é normalmente construída de maneira arbitrária, cabendo ao possuidor dos privilégios guardar os certificados e apresentá-los ao servidor quando das requisições de acesso correspondentes.

5. Integrando o SPKI / SDSI ao ambiente de gerenciamento WBEM

Nesta seção será introduzida à proposta de integração do *SPKI / SDSI* ao ambiente *WBEM* e os aspectos de implementação do protótipo.

5.1 Proposta de integração do SPKI / SDSI ao WBEM

Nas implementações *WBEM* as políticas de autorização são escritas em listas de controle de acesso (*ACLs*) locais aos servidores. Estas *ACLs* associam políticas a *namespaces* e a usuários,

não permitindo desta maneira a portabilidade das mesmas entre os diversos *namespaces* de um repositório *CIM*. Quando se considera um ambiente distribuído, a interoperabilidade de políticas entre os servidores *WBEM* de domínios distintos fica praticamente inviabilizada, pois cada servidor implementa suas próprias estruturas de controle para as *ACLs* e para os usuários. Para cada usuário deverá existir um registro nas *ACLs*, em cada *namespace* que o mesmo tiver acesso.

No modelo de interoperabilidade *WBEM* a segurança é atribuída à aplicação. Ou seja, cada servidor *WBEM* é responsável pela implementação de segurança dos objetos que protege; nos trabalhos relacionados são citadas algumas destas implementações.

Na especificação da *DMTF* [DMTF, 2003b] que define a interoperabilidade para o *WBEM*, o *CIMOM* deve estar acessível através de operações *http*; o *CIMOM* é o responsável por efetuar a autenticação e autorização dos clientes *WBEM*. Embora a *DMTF*, não defina detalhes de implementação do *CIMOM*, mesmo porque este não é o objetivo, algumas recomendações, que permitem manter a compatibilidade entre os vários fornecedores, estão especificadas, Omo por exemplo a estrutura das classes e objetos do *CIMOM* E a forma de troca de mensagens através de operações *CIM*.

Na proposta, por compatibilidade, os objetos *CIM* estão sendo representados no repositório *LDAP*, através de uma identificação única baseada no nome de classe ou *namespaces* e, em outros casos pelas propriedades chave (*KEY*) da classe. O conjunto de propriedades chave é único para todas as instâncias de classes e sub-classes. As entradas no diretório são identificadas unicamente pelo seu *DN* que também define sua localização na árvore de diretórios. O nome do objeto é formado como mostrado a Figura 2.

A identificação de principais *WBEM* está sendo feita através de chaves públicas anunciadas em certificados de identificação *SPKI*, auto-assinados. Este tipo de certificado também atribui um nome local ao principal que esta fazendo sua emissão, apenas para facilitar a memorização da identificação por parte do mesmo. Se o usuário possuir um certificado de identificação *X.509* e assim desejar, poderá publicar o certificado de nomes *SPKI*, se identificando pela mesma chave pública e pelo mesmo nome do certificado (no caso referindo-se apenas ao *CN* do nome *X.500*).

Identificadores no *CIM* são baseados em nomes e o *SPKI* é baseado em chaves. Assim, foi utilizada a classe *PublicKeyCertificate* (Figura 1) para mapear a chave pública (*PublicKey*) no respectivo nome (*Subject*) utilizado dentro de um esquema. Esta estratégia permite um certo nível de integração e portabilidade de atributos, definidos no ambiente *CIM*, para o ambiente do *SPKI*. Com base na Figura 1, pode-se perceber que na instanciação das classes *AccessControlInformation*, *Account* e *UserAccess* haverá um identificador comum (nome) que poderá ser traduzido para uma chave pública (identificador) no *SPKI* através da classe *PublicKeyCertificate* e vice-versa. Isto permite um grau de integração bastante bom em operações feitas através do *CIM* e do *SPKI*, mesmo se considerado que as estruturas de controle de cada ambiente são diferentes.

Uma alternativa para a interoperabilidade de políticas no *WBEM* poderia ser alcançada através do uso do *PCIM* (*Policy Core Information Model*) [Moore e outros, 2001] [NABHEN, R. e JAMHOUR E., 2003]. Porém, o modelo de interoperabilidade da DMTF ainda encontra-se em processo de definição, logo, não se tem maiores detalhes de como isto será considerado. Assumindo que a interoperabilidade fornecida pelo PCIM fosse adotada no *WBEM*, não haveria muita flexibilidade no esquema, uma vez que seria necessária o estabelecimento de relações de confiança inter-domínio para viabilizá-la. Tal relacionamento, compartilhando a mesma entidade de confiança, é exigido na composição das políticas de autenticação e principalmente de autorização para que as mesmas sejam aplicáveis ao um ambiente distribuído. Além disto, seria necessário relacionar em cada objeto do sistema, os

principais e os direitos associados aos mesmos nas *ACLs* do servidor dos objetos, como mostrado na Figura 5.

Um usuário com nome *X.500* de CN=UA1, tentando acessar um objeto *X.500* de CN=A1, precisa constar na *ACL* da organização A. Se este mesmo usuário (CN=UA1) precisar acessar o objeto de CN=B1, este precisa estar presente na *ACL* da organização B e, ambos devem compartilha a mesma entidade de confiança, a *CA* (Figura 5).

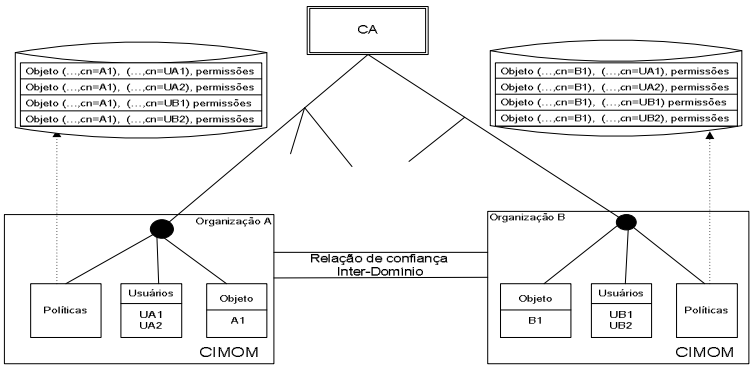

Figura 5 - Modelo *WBEM* baseado em *ACLs*

Com a integração do *SPKI / SDSI* ao modelo de gerenciamento WBEM as dificuldades citadas acima são vencidas. Com uma única entrada no repositório de *ACLs SPKI* pode ser feita a delegação de direitos de acesso a um principal (K_{U1}, Figura 6). A chave pública deste principal (K_{U1} da organização A) a partir de então poderá criar certificados de autorização que serão delegados a outros usuários (clientes *WBEM*). Se o bit delegação do certificado de autorização permitir, a autorização recebida de K_{U1} poderá ser repassada formando cadeias de autorização que distribuem a política de autorização pelo sistema. Se o *bit* de delegação estiver desativado o usuário que recebeu o direito de uma chave pública (como K_{U1}, por exemplo) poderá apenas fazer uso do direito recebido e não repassá-lo.

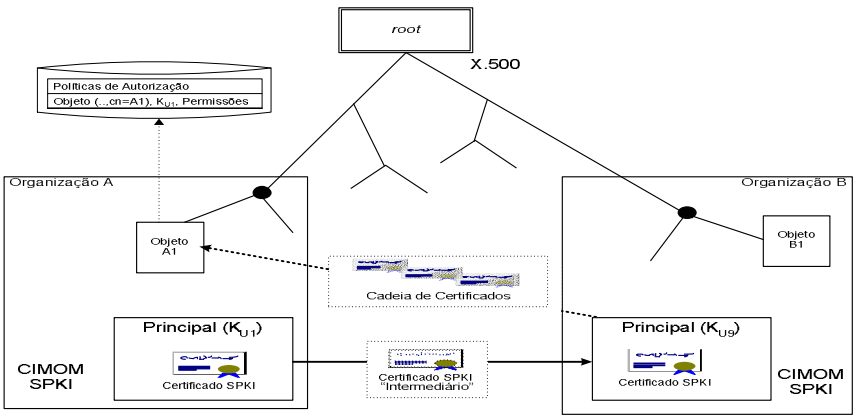

Figura 6 - Modelo de autorização SPKI / WBEM

Se a cadeia de autorização através de delegações sucessivas chegar até o principal K_{U9} (organização B, Figura 6). K_{U9} poderá utilizar a cadeia para fazer acesso ao objeto de nome *X.500* com CN=A1, sem estar nas *ACLs* da organização A, porque a cadeia de autorização lhe confere direitos para tal acesso.

Notadamente percebe-se que este esquema facilita a administração do servidor, elimina a necessidade da criação de contas em todos os domínios, onde o principal fará acessos. Além disto, elimina o ponto central de vulnerabilidades (que pode ser a entidade de confiança ou o

próprio servidor). Isto porque as relações de confiança são estabelecidas através da delegação de direitos de acesso formando as cadeias de autorização. A segurança da cadeia está baseada na assinatura digital do sistema de chave pública (*RSA*, por exemplo) aplicado a cada certificado. Como a chave que fez a delegação e a que vai receber os direitos constam do certificado assinado digitalmente, não será possível a ruptura da cadeia, pois só a chave que está recebendo o direito pode assinar a próxima delegação. A verificação da assinatura pode ser facilmente efetuada porque a chave pública do emissor do certificado de autorização consta no próprio certificado.

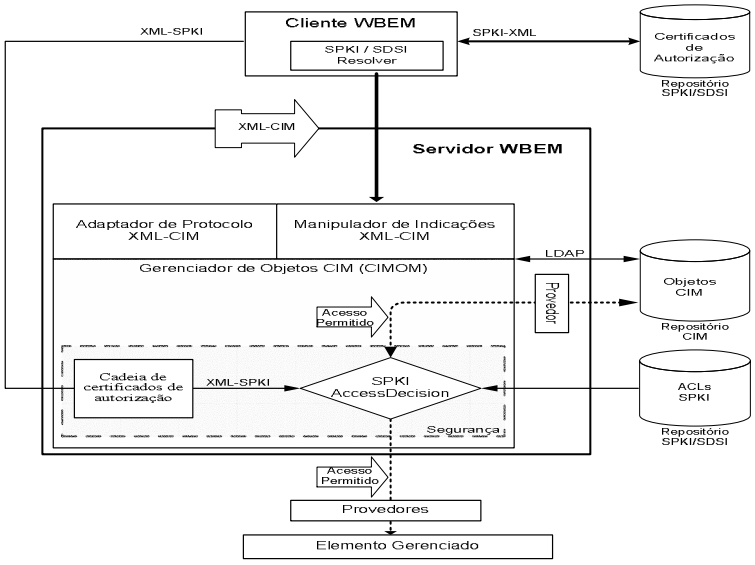

Figura 7 – Visão geral do modelo SPKI / WBEM proposto

Na Figura 7, pode-se ter uma visão geral do modelo WBEM com as extensões necessárias para suportar *SPKI* [MELLO, E. 2003]. O cliente *WBEM* tendo recebido por delegação do servidor *WBEM* os direitos que necessita para fazer uma operação, armazena o certificado em seu repositório local de certificados *SPKI / SDSI*. Então, o cliente em algum momento decide enviar uma solicitação de operação *WBEM* a um objeto *CIM* ou elemento gerenciado. O cliente monta a solicitação assinada e envia ao servidor, anexando os certificados de autorização para realizar a operação; os certificados foram recebidos anteriormente do servidor. Para recuperar os certificados da cadeia com os direitos requeridos, o cliente aciona o objeto *SPKI / SDSI Resolver* (Figura 7), que faz uma busca em seu repositório local. As conversões *XML-SPKI* e vice-versa se fazem necessárias porque o *SPKI* opera com *S-expressions* e o suporte para estruturas de dados sobre *HTTP* é provido em *XML*.

Quando a solicitação chega ao servidor *WBEM* (Figura 7), o objeto *SPKI AccessDecision* (monitor de referência) é invocado para confrontar a *ACL* recuperada do repositório de *ACLs SPKI* com a cadeia de autorização fornecida pelo cliente. Se a seqüência de certificados de autorização é válida e confere os direitos que o cliente precisa para a operação desejada o acesso é permitido, em caso contrario é negado

5.2 Aspectos da implementação do protótipo

As ferramentas adotadas na composição da arquitetura do protótipo (que implementa o modelo da Figura 7) serão consideradas a seguir.

Os protocolos *TCP* / *IP* (nativos para a Internet) foram utilizados como infra-estrutura de comunicação e para aplicação, foi adotado o *XML* sobre *http* – motivo da necessidade do adaptador *XML-CIM* e do *parser S-Expression-XML* [TERREROS, 2003] [MELLO E., 2003a].

O protótipo está baseado na implementação de referência da DMTF, *WBEMSource* (desenvolvido pelo *Open Group*) – código de domínio público [SUN, 2002] . O *WBEMSource* implementa o cliente e o servidor WBEM, em máquinas virtuais Java (JVM - *Java Virtual Machine*); no protótipo foi utilizado o *J2SE* (*Java 2 Plataform Standard Edition*) versão 1.4.1 [SUN, 1999]. Além, dos códigos do *WBEM* o *WBEMSource* oferece o compilador *MOF* para criação das classes e a implementação do repositório *CIM* (servidor *LDAP*). O repositório do cliente é um arquivo *XML*.

A biblioteca de classes *SPKI* (JSDSI2.0) utilizada no protótipo foi desenvolvida por [Morcos, 1998]. Porém, a infra-estrutura *SPKI / SDSI* e as políticas empregadas no modelo são totalmente independentes da tecnologia em uso.

No intuito de testar a implementação do protótipo foram realizadas algumas avaliações de desempenho do mesmo considerando o acesso de 1 (Tabela 1) e 10 clientes simultâneos ao servidor *WBEM* (Tabela 2). As medidas foram feitas considerando o código *WBEMSource* implementando o modelo mostrado na Figura 7 e o mesmo modelo sem o *SPKI* (como a Figura 4). Os tempos de resposta obtidos mostram que a inserção do *SPKI / SDSI* causa um certo atraso nestes tempos em relação a implementa com esquema baseado em usuário e senha (sem *SPKI*). Tal incremento no tempo de resposta se deve principalmente ao processamento necessário para efetivação e verificação da assinatura digital, no cliente e no servidor. A aplicação construída para o teste é bastante simples e permite ao cliente a requisição de acesso a um objeto no repositório de objeto gerenciado *CIM*.

Tipo de Servidor	**Tempo de resposta (ms)**
WBEM Exemplo	327
WBEM + SDSI / SPKI	432

Tabela 1 – tempo de resposta para acesso de 1 cliente

Tipo de Servidor	**Tempo de resposta (s)**
WBEM Exemplo	47
WBEM + SDSI / SPKI	52

Tabela 2 – tempo de resposta para acessos simultâneos de 10 clientes

6. Trabalhos relacionados

Várias implementações do *WBEM* foram proposta recentemente, as principais serão descritas brevemente abaixo.

Na implementação *Solaris WBEM*, o usuário é identificado por uma conta e autenticação é baseada no esquema usuário e senha. A autorização pode utilizar *Role Based Control Access* (RBAC) [Sandhu and Ferraiolo and Kuhn, 2000] deste que o protocolo utilizado seja *RMI* [Sun, 2003] , pois com *XML* sobre *HTTP* não há suporte para papéis. As políticas de controle de acesso são escritas em *ACLs* e associadas aos *namespaces*.

O *WBEMSource* [Sun, 2002], desenvolvido em plataforma Java, autentica usuário por senha e as políticas de autorização são escritas em *ACL*s associadas aos *namespaces*, para o controle, de acesso. Uma boa proposta de arquitetura para o *CIMOM* é encontrada nesta implementação.

No *pegasus* [Caldera, 2002] a autenticação é baseada em senha, mas a seção autenticada pode ser associada a um usuário ou um papel. O controle de acesso também é baseado em *namespace* e *ACL*.

Na implementação *Microsoft WMI* [Microsoft, 1999] a segurança é baseada na segurança adotada pelos servidores *Windows* e a autorização é baseada em papéis.

No *SUMO* a autenticação pode ser por usuário e senha ou através de certificados *X.509*; cada aplicação (cliente e servidor) é organizada em domínios gerenciados (*namespace*). Em cada domínio uma política de segurança deve ser definida, criando desta maneira *ACL*s garantindo a autorização no ambiente *CORBA* [OMG, 2001].

Todas as propostas relacionadas acima, a exceção do *SUMO*, baseiam seu sistema de autenticação no esquema usuário / senha. É sabido que programas como *password cracker* são

dotados de habilidade para descobrir senhas, por mais que uma política para senhas seja adotada. Em nossa proposta o mecanismo de autenticação é assinatura digital. A identificação de principais é feita através de chaves públicas e não com base no nome do usuário.

O serviço de nomes que se baseia em conta de usuário tem dificuldades de escalabilidade e necessita de relações interdomínio para operar num sistema distribuído de larga escala. Em nosso esquema não há necessidade de criação de relações interdomínio, pois as cadeias de autorização concedem direitos a chaves públicas de validade global.

Políticas de autorização escritas em *ACL* e armazenadas num servidor, tornam o servidor vulnerável. Em nossa proposta as políticas são codificadas em certificados assinados digitalmente e estão dispersas pelo sistema distribuído; o controle de acesso é feito confrontando os certificados com a *ACL SPKI* do servidor.

O modelo proposto neste trabalho é muito mais flexível e adaptado às necessidades do ambiente distribuído *WBEM* que as abordagens relatadas acima.

7. Conclusão

Considerando que o modelo de interoperabilidade *WBEM* se encontra em fase de especificação e que várias implementações referenciadas não se preocupavam muito com os aspectos de interoperabilidade. Em linhas gerais considera-se o esquema proposto mais adaptado ao ambiente *Web*.

O esquema proposto oferece uma série de vantagens em relação aos clássicos, como por exemplo: portabilidade e facilidade de administração das políticas de autorização - alcançada através das cadeias de certificados e facilidade de gerência dos usuários do sistema - alcançada pela isenção da criação de contas para os clientes do sistema.

Na proposta, a serviço de nomes pode ser eliminado e não há a necessidade do estabelecimento de relações de confiança interdomínio. A autenticação é baseada no sistema de chaves assimétrico e, portanto o problema de senhas mal formadas e dos ataques às bases de senhas é eliminado.

O teste com o protótipo mostraram a viabilidade do modelo e sua efetivação não exigiu grandes esforços de implementação no código de referência utilizado.

Referências Bibliográficas

[BÉNECH, D. and JOCTEUR-MONROZIER F. and RIVIERI, 2000]. A. Supervision of the CORBA Environment with SUMO: WBEM/CIM-Based Management Framework, CNES.

[CALDERA International Inc., 2002]. OpenWBEM Source, [online] disponível em URL : HTTP : // OpenWBEM.sourceforge.net, acesso julho de 2003.

[DMTF, 1999]. Common Information Model (CIM) Specification,Version 2.6, Agosto, 2003, URL: http://ftp.dmtf.org/cim/cimdoc20.doc

[DMTF, 2000a]. DMTF LDAP Schema for the CIM v2.4 Core Information Model, [online], disponível: HTTP://www.dmtf.org/spec/DEN/DSP0117.doc, Nov. 2000. Acesso em 01/03.

[DMTF, 2000b]. CIM User & Security Model, v2.5 disponível em [online]: *HTTP:*//www.dmtf.org/spec/*CIM*_Schema25 /*CIM*_User25.mof. Acesso em 01/2003.

[DMTF, 1999], Specification for CIM Operations over HTTP, Ver. 1.0. [online], disponível: HTTP://www.dmtf.org/download/spec/XMLs/CIMHTTPMapping10.php. Acesso em 12/2002.

[DMTF, 2003a]. Desktop Management Interface Especification, Version 2.0.1s, [online], disponivel: HTTP://www.dmtf.org/download/spec/DMI/DMI.php. Acesso em 12/2002.

[DMTF, 2003b]. Interoperability Model White Paper, CIM Version 2.7. disponivel em HTTP://www.dmtf.org/download/spec/CIM/DSP0465.DOC. Acesso em 12/2003.

[ELLISON, C. M., FRANTZ, B., LAMPSON, B., TIVEST, R., THOMAS, B. M., e YOLEN, T. 1999]. SPKI Certicate Theory. Internet Engineering Task Force RFC 2693.

[LAMPSON, B. and RIVEST, R. L. ,1996]. A simple Distributed Security Infrastructure. [online], disponivel em HTTP://theory.lcs.mit.edu/~cis/SDSI.html. Acesso 03/2003.

[M. WAHL, 1997]. A Summary of the X.500 User Schema for use with LDAPv3, RFC 2256.

[MELLO, E., 2003] Redes de confiança em sistemas de objetos CORBA, UFSC, 2003 disponível: http://www.das.ufsc.br/~emerson/mestrado.html#dissertacao. Acesso 03/2004.

[MELLO E., 2003a] - Mello, E. R., Boesel, D. F, e Carrijo, L. F. Biblioteca parserSxxS. Relatório interno DAS/cadeias de confiança, julho de 2003. Disponível em, http://www.das.ufsc.br/ seguranca/arquivos.html, acesso em março de 2004.

[MICROSOFT Corporation, 1999]. Windows Management Instrumentation: Background and Overview. Microsoft, Disponível em, htt:\www.microsoft.com\WMI, acesso em 02/2004.

[MOORE, B., ELLESON E., STRASSNER J. AND WESTERINEN A., 2001]. Policy Core Information Mode. RFC3060l, IETF.

[MORCOS, A. 1998]. A Java implementation of Simple Distributed Security Infrastructure. Master's thesis, MIT.

[NABHEN, R. e JAMHOUR E., 2003] RBPIM: A PCIM-Based Framework for RBAC, disponível: http://www.ppgia.pucpr.br/~jamhour/download/outros/artigos/RBPIMFinal.doc, acesso em março de 2004.

[OMG, 2001]. The Common Object Request Broker: Architecture and Specification, Editorial Revision: CORBA 2.4.2, OMG: Fevereiro 2001

[RIVEST,R.L, 1997]. S-expressions. [online], disponível em, http://theory.lcs.mit.edu/~rivest/sexp.html. Acesso em 06/2003.

[SANDHU, R. and FERRAIOLO D. F. and KUHN R. , 2000]. The NIST Model for Role Based Access Control: Towards a Unified Standard, Proceedings, 5th ACM Workshop RBAC.

[STALLINGS, 1994]. William – SNMP, SNMPv2, and CMIP, Addison-Wesley.

[SUN Microsystems Inc, 1999]. WBEM on Sun Developer's Guide, Palo Alto California.

[SUN Microsystems Inc, 2002]. WBEM Services. [online], disponivel em *HTTP*://WBEMservices.sourceforge.net. Acesso em janeiro de 2003.

[SUN Microsystems Inc, 2003]. The Java Tutorial. [online], disponivel em java.sun.com/docs/books/tutorial/index.html.

[TERREROS, Xavier Orri Sainz de los, RIBES, Joan-Maria Mas, 2002]. SPKI-XML Certificate Structure. [online] Disponível em http://www.oasis-open.org/cover/xml-spki.html. Acesso em janeiro de 2003.

[T. HOWES, M. WAHL, S. KILLE, 1997]. Lightweight Directory Access Protocol, RFC2251.