2
Trabalhos Relacionados

Este capítulo apresenta trabalhos relacionados ao problema da travessia de firewalls/NAT por aplicações CORBA, alguns dos quais tiveram grande influência no desenvolvimento desta dissertação.

2.1
OMG

Em 2004 a OMG (Object Management Group) publicou uma especificação [5] para tentar resolver o problema da travessia de firewalls por aplicações CORBA. Esta especificação tem como principal objetivo fazer com que aplicações CORBA atravessem firewalls de uma maneira padronizada, procurando causar o mínimo de processamento e de complexidade nsa ações realizadas pelo firewall.

Antes de proceder com o detalhamento da especificação, é necessário apresentar uma classificação de firewalls introduzida por esta última. A especificação classifica os firewalls como sendo ou um **Firewall TCP** ou um **Procurador de Aplicação**, de acordo com o nível de protocolo no qual ele atua. O primeiro indica o firewall mais comum e genérico, o qual interfere em protocolos no nível de transporte (TCP ou UDP normalmente) e controla o acesso de/para a rede interna através de mapeamentos de endereços IP e portas válidos. Para uma conexão (ou mais genericamente, um pacote) ter sua passagem permitida pelo firewall, é necessário que tenha sido cadastrado no firewall um mapeamento associando o endereço/porta de origem com o endereço/porta de destino indicado no pacote. Não existe neste tipo de firewall nenhum tipo de processamento no nível de aplicação. Sendo este tipo de firewall mais comum e mais genérico, geralmente é utilizado por uma grande variedade de aplicações. Sua principal desvantagem é a espessa granulosidade

de interceptação, controlando o acesso apenas no nível de endereço/porta.

O segundo tipo de firewall (procurador de aplicação), como o próprio nome indica, atua no nível de aplicação, podendo assim tomar decisões de acesso com uma granularidade mais fina (por exemplo em nível de objetos e métodos). Este tipo de firewall é menos genérico de que o firewall TCP uma vez que cada aplicação deve desenvolver o seu procurador de aplicação (ou adicionar funcionalidades a um firewall existente). A partir deste ponto, o termo "procurador de aplicação" será utilzado para indicar um firewall que trata o protocolo IIOP, visto que é o protocolo de aplicação que interessa ao presente trabalho. Portanto, este elemento deve ser capaz de interceptar e processar as informações contidas em um pacote IIOP, a fim de permitir ou não o encaminhamento de pacotes.

A proposta da OMG consiste na introdução de estruturas de dados no perfil IIOP do IOR e de uma nova mensagem GIOP. Todo ORB que estiver protegido por um ou mais firewalls (de qualquer tipo) e hospedar um objeto que deva ser alcançado por elementos situados na rede externa, deve construir um componente rotulado (*Tagged Component* [6]) chamado `TAG_FIREWALL_PATH`. Neste componente existem informações sobre todos os elementos (firewalls) existentes em um determinado caminho entre a rede externa e o ORB, incluindo este. Estas informações incluem tanto os endereços dos elementos como o serviço oferecido por eles (tunelamento IIOP, TLS, SecIOP, ...). Pode haver mais de um componente deste tipo (indicando assim que existe mais de um caminho) e eles devem ser inseridos na sequência de componentes rotulados existente no perfil IIOP do IOR[1], que será exportado para os elementos da rede externa. A forma como o ORB deve ser configurado com essas informações não é definida pela especificação, sendo assim um detalhe de implementação. Apenas é definida a semântica e a sintaxe do componente rotulado `TAG_FIREWALL_PATH`.

Uma vez que um ORB cliente tenha recebido um IOR com este componente, ele tem a opção de se conectar diretamente ao ORB do objeto CORBA utilizando o endereço contido no perfil IIOP ou utilizando as informações contidas no componente rotulado `TAG_FIREWALL_PATH`. Este processo de decisão é dependente de implementação, caso se opte pelo uso do componente rotulado, então o ORB cliente deve construir uma mensagem introduzida pela especificação chamanda `NegotiateSession` antes de enviar a mensagem de requi-

[1] O perfil IIOP do IOR contém um conjunto de componentes rotulados [1] os quais podem prover informações sobre o objeto em questão como por exemplo travessia de firewall, política de transação, trasporte SSL, entre outros

sição GIOP . Esta nova mensagem contém uma estrutura de dados chamada `FIREWALL_PATH` como uma entrada de contexto de serviço [6], que por sua vez deve conter um caminho entre o ORB cliente e o ORB servidor. Os elementos (firewalls) que por ventura existam entre o ORB cliente e a rede externa também devem estar contidos na mensagem, e a maneira como essa informação deve ser fornecida ao ORB cliente é dependente de implementação.

Uma vez contruída esta mensagem, ela é enviada ao primeiro elemento do caminho, que irá processá-la e encaminhá-la ao elemento seguinte e assim por diante. No caso de firewalls TCP, por eles não atuarem no nível de aplicação, a mensagem GIOP não é examinada e portanto devem haver mapeamentos previamente configurados para liberar a passagem da mensagem. No caso de procuradores de aplicação a mensagem é analisada e processada conforme descrito em [5]. Este processamento visa verificar, entre outras coisas, se a conexão é permitida e pode assim ser encaminhada ao próximo elemento, além de atualizar dados da própria mensagem. Este processo continua até o último elemento antes do ORB servidor que seja capaz de entender o protocolo IIOP (i.e. ORBs ou procuradores de aplicação) que, ao identificar ser ele o último, constrói uma mensagem de resposta e a envia pelo caminho inverso. Esta mensagem eventualmente chegará ao ORB cliente que, em caso de sucesso, poderá enviar mensagens GIOP normais pela sessão aberta.

A figura 2.1 ilustra um caso de uso da solução apresentada pela especificação. Nela um objeto CORBA identificado pela letra F está protegido por três firewalls, sendo dois firewalls TCP (elementos C e E) e um procurador de aplicação (elemento D). Um ORB cliente (identificado pela letra A) por sua vez está protegido por um procurador de aplicação (identificado pela letra B). Assumimos que o ORB responsável pelo objeto CORBA tenha sido previamente configurado com a informação dos elementos entre ele e a rede externa; a forma como essa informação é transferida é dependente da implementação. Ao gerar o IOR, é inserido na sequência de componentes rotulados o componente relativo à travessia de firewalls contendo a sequência de firewalls entre a rede externa e o ORB na ordem correspondente (elementos C, D, E e F). Ao receber este IOR, o ORB cliente A decide contactar o objeto não diretamente, mas através da sequência de firewalls. Para isso, ele cria a mensagem GIOP `NegotiateSession` contendo toda a cadeia ordenada de elementos entre ele e o ORB desejado, incluindo este, (elementos B, C, D, E e F) e a envia ao procurador B, pois é ele o primeiro da cadeia. O procurador de aplicação B analisa a mensagem, atualiza seus dados conforme descrito em [5] e a encaminha ao firewall TCP C, que deve ter uma regra mapeando o seu endereço

com o endereço do procurador de aplicação D. O procurador de aplicação D analisa a mensagem e identifica ser ele o último elemento capaz de compreender o protocolo IIOP no caminho para o ORB servidor. Caso seja permitida, a conexão com o próximo elemento é feita, nesse caso com o firewall TCP E que possui um mapeamento para o ORB F, e uma mensagem de resposta positiva é construída e enviada pelo caminho inverso. Caso a conexão não seja permitida é enviada uma exceção e o processo é finalizado. Quando a resposta positiva chegar ao cliente o mesmo possui uma conexão aberta com o ORB servidor, por onde pode enviar qualquer mensagem GIOP.

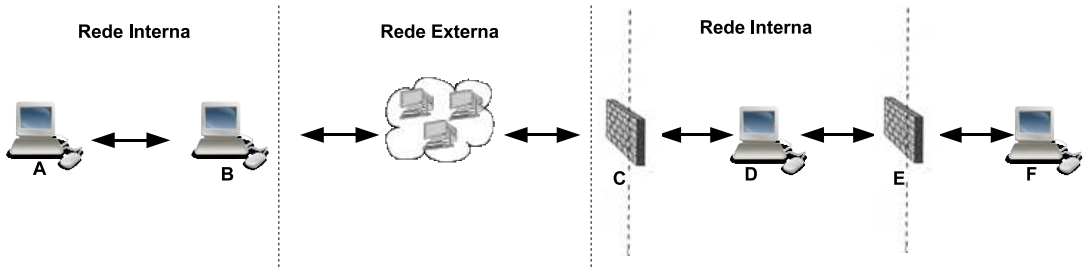

Figura 2.1: Caso de Uso da Especificação de Travessia de Firewalls da OMG

Apesar de procurar ser uniforme e impor o mínimo de processamento por parte do firewall, a solução apresentada pela OMG requer modificações tanto no ORB do lado do cliente quanto no ORB responsável pelo objeto servidor, além de exigir a configuração de firewalls TCP e a construção dos procuradores de aplicação.

2.2 JXTA

JXTA [7] [8] é um projeto de código aberto, criado inicialmente pela Sun Microsystems e depois aberto à comunidade de desenvolvedores em geral, que visa oferecer serviços e infra-estrutura para o desenvolvimento de aplicações ponto-a-ponto (*peer-to-peer* - P2P). Ele consiste de um conjunto de protocolos que permite que estações formem uma rede virtual *ad hoc* capaz de se auto-organizar em grupos, independente de sua localização na rede e de uma infra-estrutura centralizada de gerenciamento. Os pares podem assim oferecer serviços uns aos outros utilizando a rede JXTA como infra-estrutura de comunicação..

Apesar de não utilizar CORBA como meio de comunicação, o interesse deste trabalho no projeto JXTA é que o mesmo também provê uma solução para a travessia de firewall/NAT. Isto é conseguido através de tunelamento HTTP e do uso de pares especiais, que funcionam como procuradores e são chamados de *relay peers*. O par que estiver protegido por um firewall e/ou NAT e quiser entrar na rede JXTA, indica um procurador (*relay peer*). Na Internet existem procuradores "públicos" mantidos pelo próprio projeto, apesar de nada impedir de um usuário criar um. A partir de então todas as mensagens destinadas ao par protegido, serão direcionadas para o procurador que as armazenará até que o par protegido consulte o procurador via uma requisição HTTP para saber se há mensagens para este. [2]. Em caso positivo, o procurador envia a(s) mensagem(ns) ao par protegido através de uma resposta HTTP.

Essa solução tem como vantagem o fato de não requerer qualquer configuração no elementos de firewall e NAT para que possa funcionar. Seu único requisito é que estes últimos permitam que elementos internos iniciem o protocolo HTTP para fora da rede, o que geralmente é permitido. Além de não dar suporte à CORBA, esta solução tem como desvantagem a consulta periódica por parte do par protegido ao procurador que desperdiçará largura de banda, caso seja muito frequente, ou caso contrário, irá impor um atraso excessivo à aplicação.

2.3 ICE

A plataforma ICE [9] é um middleware de comunicação similar conceitualmente com CORBA sendo, segundo os seus autores, mais simples e poderosa. Ela provê uma solução para travessia de firewalls e NAT através de um serviço chamado Glacier [10]. Este serviço deve assumir o papel de firewall da rede, mas pode funcionar em conjunto com outro firewall. Seu princípio de funcionamento básico é isolar tanto o cliente quanto o servidor um do outro. O cliente deve estar configurado para utilizar o Glacier como servidor, e o servidor interpreta o Glacier como sendo o seu cliente, ou seja, o Glacier funciona como um corretor (*broker*) entre os dois.

Assim como a solução apresentada na seção 2.2, esta não dá suporte a CORBA além de requerer a configuração ou substituição do firewall da rede.

[2] O procurador é portanto também um servidor HTTP

Para o servidor a solução é transparente, havendo apenas a necessidade de configurar o cliente para utilizar o Glacier.

2.4 Xtradyne

A Xtradyne é uma empresa que possui um produto comercial chamado I-DBC (*IIOP Domain Bondary Controller*) [11]. Este produto é um firewall capaz de tratar o protocolo GIOP/IIOP e basicamente o seu princípio de funcionamento é substituir no perfil IIOP dos IORs que passam por ele o endereço do ORB servidor pelo seu próprio endereço. Outra função desse produto é identificar IORs passados como parâmetros em invocações CORBA e realizar a substituição descrita anteriormente.

Esta solução tem como grande vantagem a transparência em relação a clientes e servidores CORBA: nenhum dos dois precisa ser modificado ou configurado para utilizá-la. A única exigência é a alteração do IOR gerado, para que o endereço original do objeto CORBA no perfil IIOP seja substituído por um endereço do firewall. Além de ser comercial e exigir a aquisição de licença, esta solução também requer a configuração do firewall da rede.