Grupo № 23



SISTEMAS DISTRIBUÍDOS

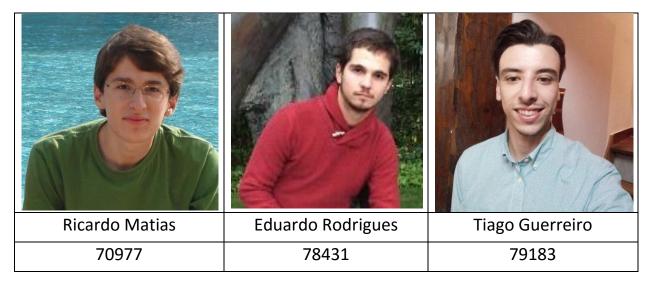
2.º Semestre 2015/2016

PROJECTO Relatório de Projecto

Grupo: A67

URL do Repositório: https://github.com/tecnico-distsys/A_67-project

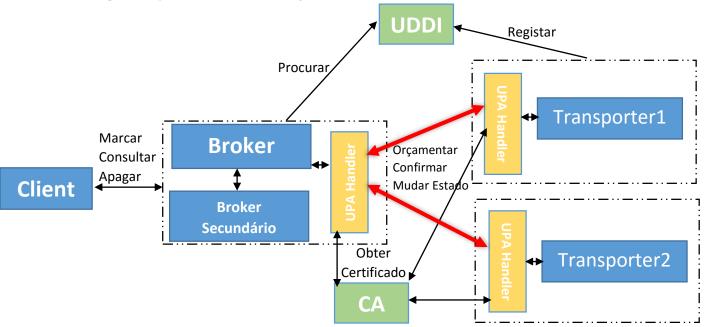
Realizado por:



1 Segurança

Permite garantir a integridade dos dados trocados entre Transporter e Broker e vice-versa. Com a solução apresentada as mensagens garantem: frescura, autenticação e não repúdio.

1.1 Figura Representativa da Solução



1.2 Descrição da Figura

Entidade/Objecto	Função/Explicação	
UPAHandler	Corresponde a uma classe que após apanhar SOAPMessages, verifica se as mesmas estão correctas ou adiciona segurança através da alteração das mensagens.	
CA	Entidade máxima e segura que atribui certificados.	
↔	As mensagens que circulam neste canal(Broker <-> Transporter) têm de ser protegidas.	

1.3 Explicação Sucinta da Solução Executada

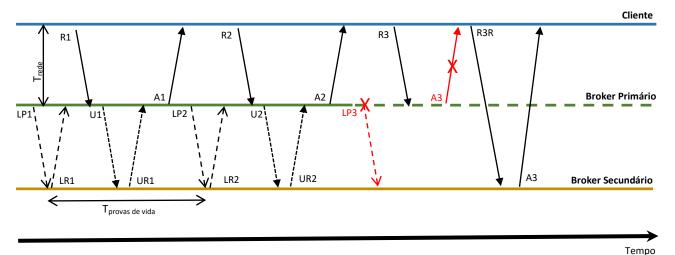
A CA é contactada pelo receptor para verificação do certificado de um emissor, garante que certificado é confiável. Contacto apenas de 30 em 30 pedidos, garante que certificado é actualizado e não foi revogado, assegura também que a CA não é sobrecarregada. <u>Exemplo:</u>

P	Mensagem a sair de Broker	Garante	Mensagem a entrar em Transporter
	Adicionado Header com:		Mensagem recusada se:
1	Sender <<i>UpaBroker</i>>		Após remover assinatura, e verificar a
2	Nounce <timestamp></timestamp>	Frescura	mensagem com a Kpública do emissor
3	Signature < caracters > Estes caracters correspondem a um digest de toda a mensagem SOAP And Andread Source 2 diffrade company of the source of the s	Não repúdio Autenticidade	 não houver match. Algum elemento não presente no header ou header vazio. Nounce já recebido ou com mais do
	produzida até ao Passo 2 cifrada com a Kprivada do Sender.		 Nounce já recebido ou com mais de 30 segundos.

2 Replicação – Tolerância a Faltas

Garantir a continuidade do modelo de funcionamento síncrono do serviço Broker após falha silenciosa permanente deste com recurso a um servidor Broker Secundário.

2.1 Figura Representativa da Solução



2.2 Descrição Parcial da Figura

Os seguintes passos descrevem a figura:

- Assumimos que o tempo de envio da mensagem na rede (T_{rede}) é de 2 segundos.
- BP (Broker Primário) envia provas de vida (LP) a cada 3 segundos.
- BS (Broker Secundário) responde à prova de vida (LR).
- Cliente faz pedidos (R) ao BP.
- Durante o processamento do pedido o BP, actualiza (U) o BS.
- Após receber a resposta do BS (UR), o BP envia a resposta ao pedido ao Cliente (A).
- *Falha silenciosa* no BP. Prova de vida (LP3) não enviada ao BS.
- Cliente faz um pedido(R3) ao BP e não irá obter resposta (A3). Passado (T_{provas de vida} + T_{rede}), o BS detecta a falha do BP e regista-se no UDDI com o endereço do BP.
- Cliente volta a repetir o pedido (R3R), comunicando com o BS e obtém a resposta (A3).

2.3 Explicação Sucinta da Solução Executada

Para resolvermos o problema, decidimos implementar replicação activa no Broker. O Broker possui agora um servidor secundário que guarda todas as actualizações feitas no servidor primário. Sempre que o servidor primário recebe um pedido do cliente, envia um pedido ao secundário para actualizar o mapa com a resposta das transportadoras e após saber que o secundário guardou a informação, actualiza o seu próprio mapa e responde ao cliente. O cliente reenvia o pedido sempre que não obtenha resposta. Para garantir a funcionalidade do sistema, o servidor secundário recebe provas de vida do servidor primário a cada 3 segundos. Se não receber nenhuma prova de vida durante (T_{provas de vida} + T_{rede}), o secundário assume-se como primário, regista-se no UDDI com o endereço do servidor primário e liga-se a todas as transportadoras. Se, durante este passo, o cliente fizer pedidos ao Broker, irá ser lançada uma excepção (não encontrou o Broker).