

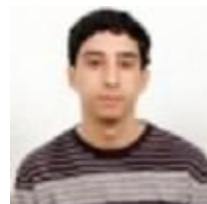
# Sistemas Distribuídos 2015-2016

## Grupo A61

Leo Francisco Clément Valente    nº 67030



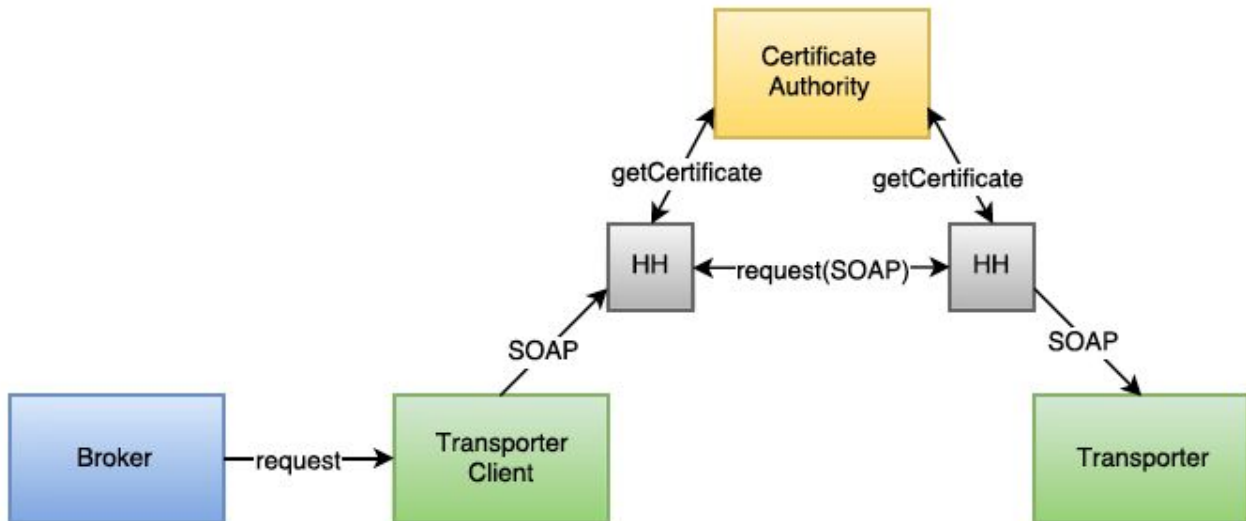
Duarte Miguel Tavares Barreira    nº 64752



João Daniel Cardador Feneja    nº 73978



# 1. Segurança



## 1.1. Descrição da Figura

A figura retrata o modelo de segurança no nosso projecto. O **Broker** faz *request* ao **Transporter Client**, que comunica com o **Transporter** através do protocolo *SOAP*. Os **HH (Header Handlers)** comportam-se nesta comunicação como *Man in the middle*, isto é, vão interceptar a mensagem que é enviada/recebida. O **Certificate Authority** distribui certificados para verificar a autenticidade das mensagens *SOAP*.

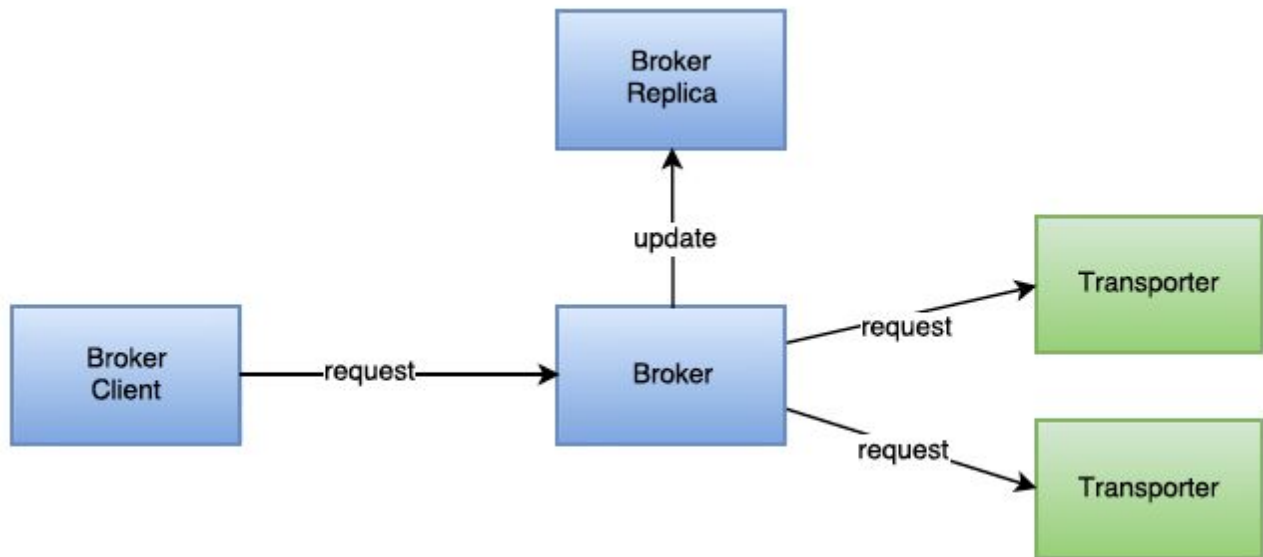
## 1.2. Explicação da Solução

Para garantir que a mensagem é autêntica e que não foi alterada vamos usar a chave privada do emissor para cifrar o resumo (*digest*) do *Body* da *SOAP Message* enviada e vamos inserir esse resultado no *Header* da *SOAP Message* no campo 'signature'.

Isto significa que quem recebe a mensagem terá a garantia que a mensagem é autêntica se decifrar o *Header* com a chave pública do emissor e comparar com o *digest* do *Body*. Isto garante também o não-repúdio.

Para garantir a frescura é inserido no *Header* (no campo 'timestamp') a hora que esta mensagem foi enviada, cifrada da mesma forma que o *digest* do *body*. Existe uma tolerância de 2 segundos para que deixe de se considerar como fresca uma mensagem.

## 2. Replicação



### 2.1. Descrição da Figura

A figura retrata o modelo de replicação no nosso projecto. O **Broker Client** ao fazer *request* ao **Broker**, este irá comunicar com os **Transporters** a fim de ver orçamentos e o estado das viagens. No fim de ter toda a informação e apenas caso o trabalho seja aceite ou rejeitado, o **Broker** irá fazer *update* no **Broker Replica**, garantindo deste modo que a réplica está sempre actualizada e pronta a intervir em caso de falha. O **Broker Replica** também faz tarefa de *clear* quando o servidor principal o faz.

### 2.2. Explicação da Solução

Para garantir que em caso de falha do **Broker** o programa continue a funcionar, o **Broker Replica** vai sendo actualizado constantemente e pede provas de vida ao servidor **Broker** de 10 em 10 segundos. Depois de os **Transporters** enviarem a informação pedida o **Broker** faz *update* no **Broker Replica**. Caso haja uma falha no **Broker**, a réplica toma o seu lugar, desde logo sabendo qual o ponto da situação.

O **Broker Replica** toma o lugar do **Broker** em 2 situações:

- *Timeout*: **Broker** não conseguiu responder a tempo pelo que não deu provas de vida e lançou uma excepção.
- *Lookup* no **uddi** devolveu *null*: **Broker** terminou a sua execução graciosamente, deixando lugar à réplica.