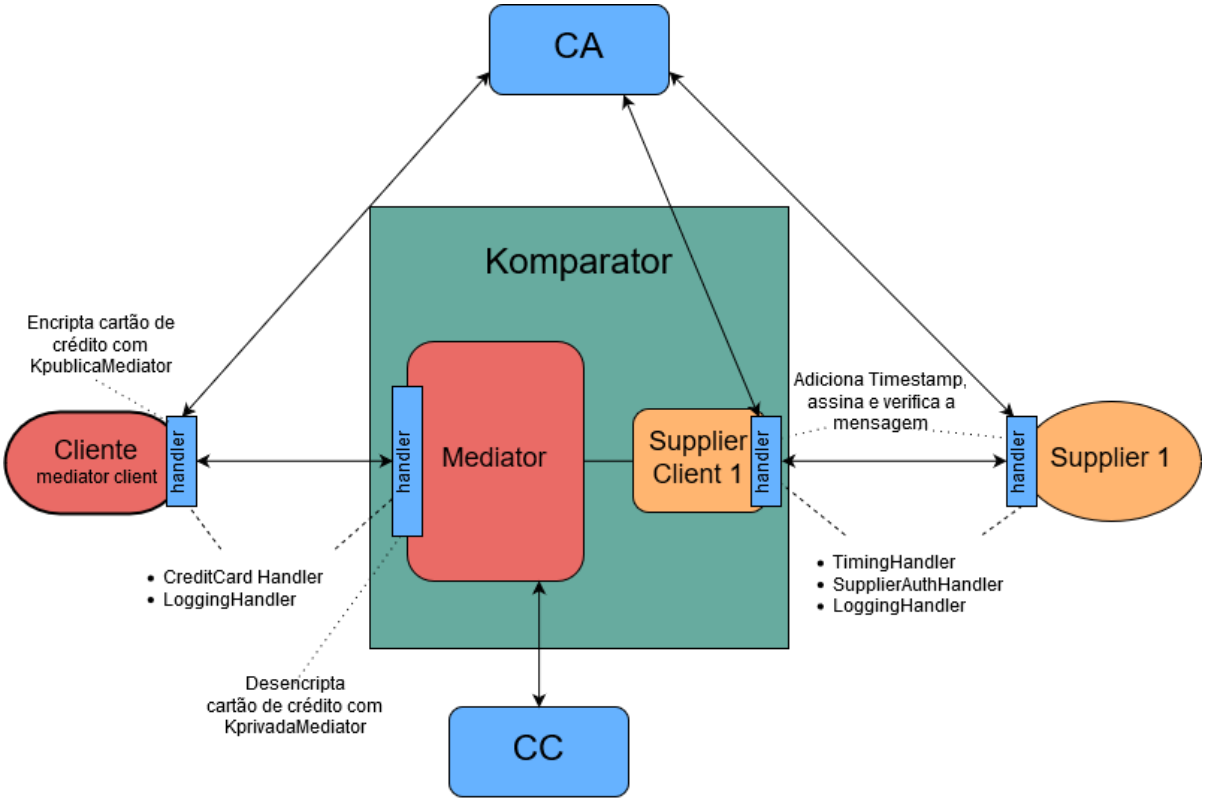
Komparator

A52

		
Daniel Correia 80967	João Crespo 81811	Pedro Caldeira 81888
https://github.com/tecnico-distsys/A52-Komparator		



Requisitos de segurança do problema

1) Confidencialidade da informação sensível relativa ao cartão de crédito.

A nossa solução consiste em **implementar um handler que usa um sistema de cifra assimétrica**. O módulo MediatorClient começa por obter a chave pública do Mediator para de seguida encriptar o número de cartão de crédito. **Assim, apenas o Mediator conseguirá decifrar o número de cartão de crédito com a sua chave privada e obter o seu valor original.**

2) Autenticidade e integridade das comunicações entre Suppliers e Supplier Clients.

Implementámos outro handler que assina a mensagem SOAP que intercepta. Este handler adiciona à mensagem a assinatura do remetente e um resumo da mensagem, obtido com o algoritmo SHA256 e cifrada depois com a chave privada do remetente. Quando o destinatário recebe a mensagem, começa por pedir ao CA a chave pública do remetente e usa-a para decifrar a assinatura. **Se o resultado obtido for igual ao resumo da mensagem, a mensagem não foi modificada e foi enviada pelo remetente certo, garantindo assim integridade e autenticidade.**

3) Frescura das comunicações entre Suppliers e Supplier Clients.

Para satisfazer este requisito **implementámos um handler que adiciona um timestamp a cada mensagem SOAP. Quando um destinatário recebe uma mensagem, verifica se foi enviada há mais de 3 segundos ou no futuro**. Caso isto aconteça, a mensagem é recusada e a resposta é contém uma SOAPFault com o erro (RuntimeException).

No entanto, com este método, por mais baixo que seja o intervalo temporal imposto, a mensagem continua sujeita a um *replay attack*. Para contrariar este ataque a frescura devia ser garantida com um nonce. O Emissor coloca um *nonce* (random number com mistura de timestamp) na mensagem que envia (O nonce vem na assinatura da mensagem). O Recetor deve ter uma lista onde guarda os *nonce*'s e verificar se já recebeu alguma mensagem com esse nonce. Se sim, quer dizer que está a acontecer um *replay attack* e a mensagem não deve ser aceite.

Estrutura das Mensagens SOAP com a solução de segurança

Mensagem enviada pelo Mediator-Client para o Mediator na operação buyCart

```
<S:Envelope><SOAP-ENV:Header/><S:Body>
      <ns2:buyCart>
      <cartId>cart1</cartId>
      <creditCardNr>hGwcKe0DIpDFOo2gYkk
      iQawb7at1RN145NweeRRSSMnkkamqUfRr
      jXKJ3jaamIptr+3Me8KxtaXTutTjyDtGm
      </creditCardNr>
    </ns2:buyCart>
  </S:Body>
</S:Envelope>
```

Número de cartão de crédito encriptado, garante confidencialidade.

Mensagem enviada pelo Supplier para o Supplier-Client na operação buyCart

TimeStamp da mensagem, garante frescura

Nome do serviço, necessário para o destinatário obter a chave pública do remetente e verificar a assinatura

Assinatura da mensagem, garante autenticidade e integridade

```
<S:Envelope>
  <SOAP-ENV:Header>
    <w:timeStamp>2017-05-04 23:11:40:619</w:timeStamp>
    <w:serviceName >A52_Supplier1</w:serviceName>
    <w:signature>gZIUDaqCPHJOeA5jxYDzHCi2j2bGNS25aIyn2H
    cgWY1Z+9VKGpDgO4Iu0zwEoK/rkjfBdKp1i+n5RzuVNvruI1gI1
    XKs1QaPYOOwnvXBCCP6h2sZ1h3K0usolFuF8D7r88hE4p9AzufR
    7Ag==
    </w:signature>
  </SOAP-ENV:Header>
  <S:Body>
    <ns2:getProductResponse >
      <product>
        <id>X1</id>
        <desc>Basketball</desc>
        <quantity>0</quantity>
        <price>10</price>
      </product>
    </ns2:getProductResponse>
  </S:Body>
</S:Envelope>
```