Security 2018/2019

Auction System

8240 - INTEGRATED MASTER IN COMPUTER ENGINEERING AND TELEMATICS

Filipe Reis
NMec: 76414 filipereis96@ua.pt

Carlos Ribeiro
NMec: 76771
carlosfiliperibeiro@ua.pt

16 November 2018

Contents

1 Introduction 3

2 Requirements 4

3 System Components 4

4 Mechanisms 5

4.1 Key Distribuition 5

4.2 Cipher 5

4.3 Message Authentication Codes 5

4.4 Signatures 5

4.5 Certificates 6

5 Working pipeline 6

5.1 Establish a session 6

5.2 Message exchange 6

5.3 Client 6

5.4 Receipts 7

6 Conclusion 8

1 Introduction

The purposed project is called "Blockchain-based auction management". The main goal is to build an auction manager safe, to that we will implement a set of security policies, for example, protect the messages exchanged between the different entities, identify a bid author with CC, validate a receipt of each bid, etc. A auction can have different bidding types, for example, an English or blind auction. The first one is based on the current bid to be higher than the last one. The second one in bidding without knowing what the other clients do, who has bid higher wins the auction. In our solution, we are going to use an English auction style.

2 Requirements

The system requires:

- **Bids confidentiality, integrity and authentication.** Bids cannot be modified or forged.
- **Bid acceptance control and confirmation.** Bids are only accepted if they fulfill the English auction style.
- **Bid author identity and anonymity.** Bids are linked to users by their CC, however they shall remain anonymous until the end of a auction.
- **Honesty assurance.** The auction repository shall have access to all the information about the bids, but shall not be able to act differently according with the client.

3 System Components

The system will be based in 3 parts, which are :

- **Auction Manager** Server where all the information will be managed and sent to their place.
- **Auction Repository** Server with all the information about the auctions.
- **Clients** Normal auction client that is able to do a set of instructions.

4 Mechanisms

4.1 Key Distribuition

In order for the client communicate with the Auction Manager, and the Manager communicate with the Auction Repository the pairs have to agree a key.

Diffie-Hellman is the algorithm we found to do the key agreement, it creates a safe channel between two peers in such a way that the secret can't be seen by observing the communication.

4.2 Cipher

Our system is goign to use symetric block ciphers.

For that we are using AES algorithm with Cipher Block Chaining (CBC) mode.

4.3 Message Authentication Codes

Message authentication codes(MAC) are used to provide information integrity and authenticity. A MAC is calculated with a key and the message we want to authenticate.

4.4 Signatures

We need to be able to ensure a message authenticity, MAC helps with it but it does not completely solves the problem, so we are also going to use signatures in order to do it.

4.5 Certificates

A digital certificate allows other parties to rely on signatures made about the private key that corresponds to the certified public key. This will be used to validate signatures.

5 Working pipeline

5.1 Establish a session

All the messages are going to be encrypted using AES, they are also going to have an associated MAC, to that the server and the client must agree a shared key by both, this is where Diffie-Hellman algorithm enters in action by making the key agreement.

5.2 Message exchange

Once the session is established to send a message it must be ciphered with the key provided by Diffie-Hellman algorithm, after that it is calculated his MAC with the messaged already encrypted, we do that to save resources, and finally it is sent, once received on the server side, the MAC is validated, if it is valid than it is deciphered with the same key. As this system is based in a blockchain and we need to be able to order the bids, each bid will be encrypted with the last one hash.

5.3 Client

In order to be able to authenticate a message, signatures shall be used, as CC uses PKCS1 we are also going to use that algorithm. A certificate shall be exchanged between client and server so server knows he can trust client, CC uses asymmetric keys,RSA, so server shall also use it.

5.4 Receipts

To create a receipt, the bid shall pass by a hash function, than concatenated with a time stamp and than pass again by a hash function, after all that it is ciphred with the authentication key from CC.

6 Conclusion

We have found some points that we haven't understood, like how Cryptopuzzles or Certificates will work on this project. Although, we haven't thought properly what and how we are going to develop this project, we already have a very basic ideia and we believe we will be able to conclude this project sucessfully.