TÉCNICO LISBOA

Instituto Superior Técnico

==========

SEC Report

==========

Group 23

André Fonseca 84698
Leonor Loureiro 84736
Sebastião Amaro 84767

Professor: Nuno Anselmo
Date: 11/04/2019

Possible threats:

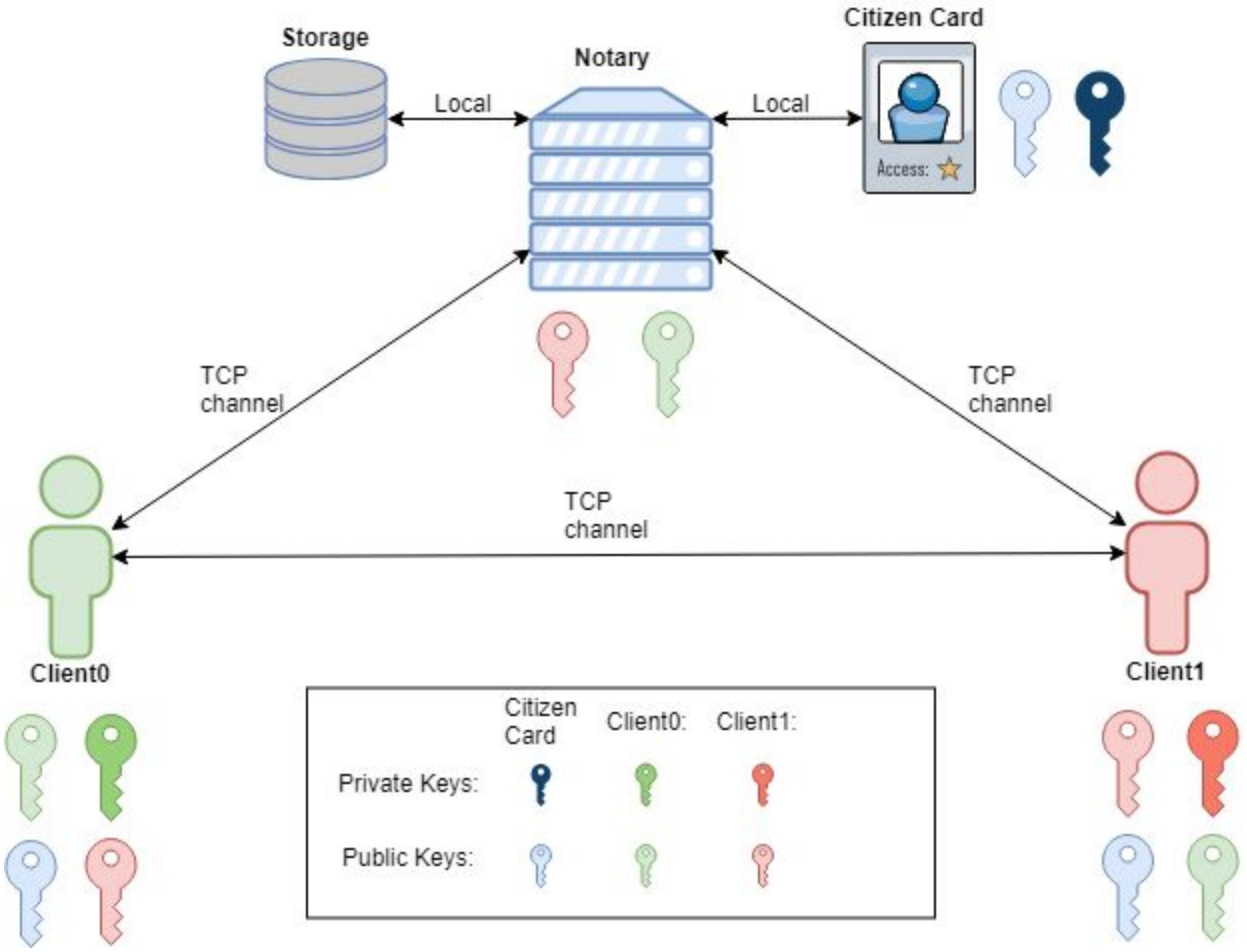
Unauthorized insertion of forged information and modification of information in transit.
Usually exploited through attacks such as **Man in the Middle**.

These threats are avoided by adding a **digital signature** and **freshness tokens** to all messages exchanged, thus providing **integrity** and **authentication,** allowing the detection of modifications made to the original message and discarding any unauthorized request/response.

Unauthorized replay of information (replay attacks)
It is prevented through the use of a **nonce** and a **timestamp**, which guarantees **freshness**. This way, messages that have already previously been seen are disregarded.

Service Overloading
Commonly done through DOS and DDOS attacks.
Such attacks are not being dealt with, but could be diminished by a firewall and infrastructure redundancy.

Design:



Key Distribution

The **Notary Server** contains:

Access to a citizen card with a public and private key.
Every client's public key.

The **Clients** contain:

Notary's citizen card public key.
Every client's public key.

Communication

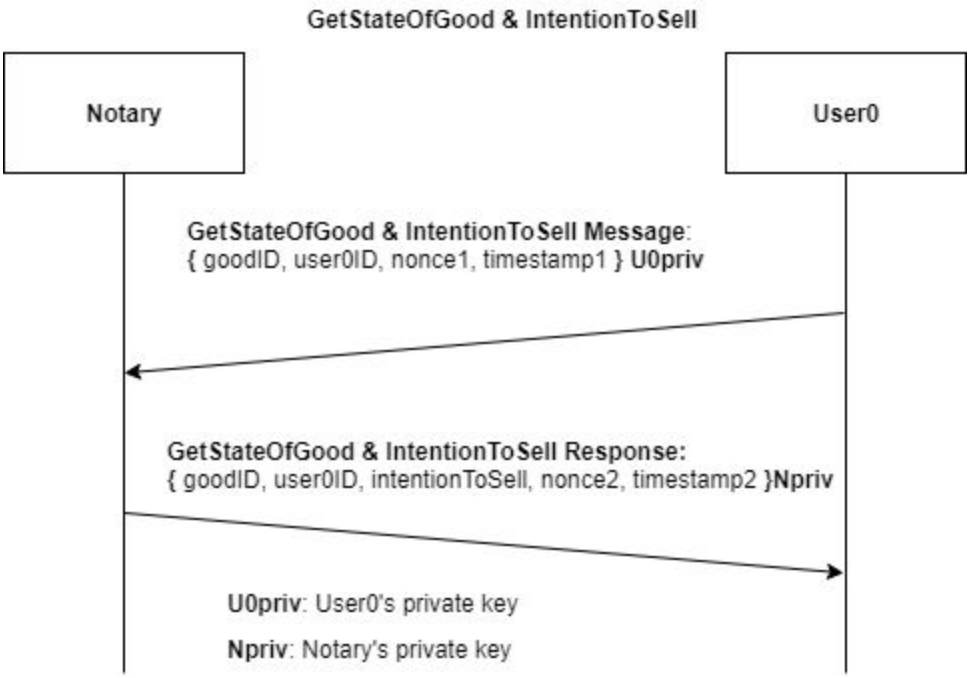
The communication between clients and client server is done through **TCP sockets**.
Sending Messages:

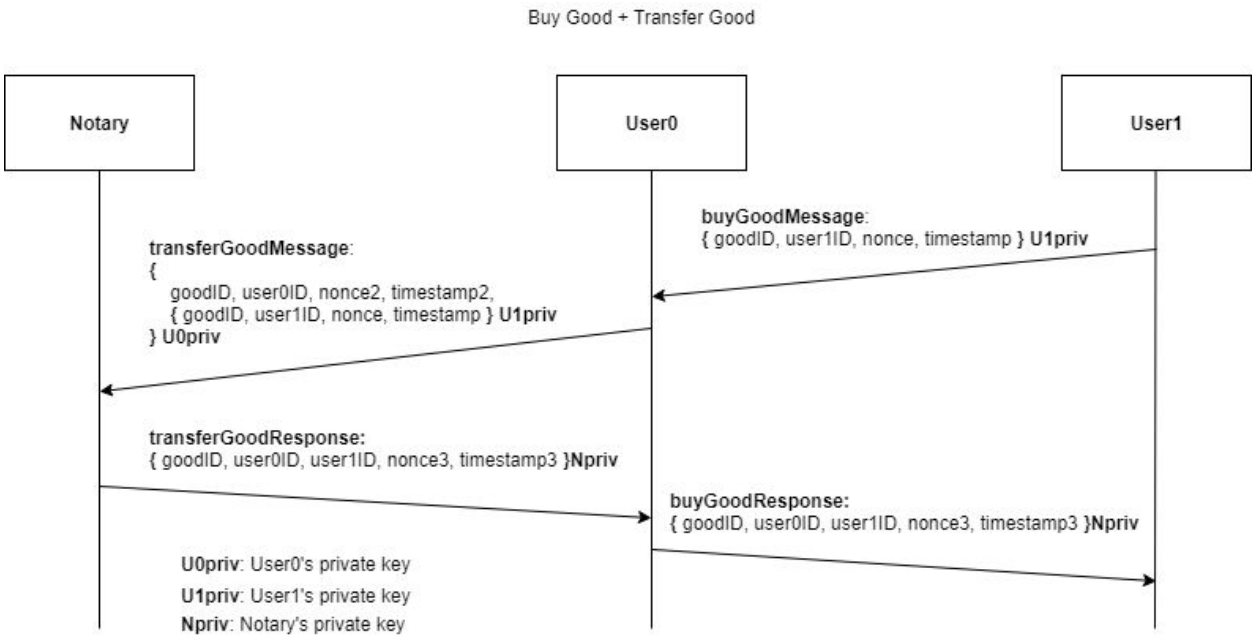
Each message has an **operation identifier** and the operation's **arguments.**
Every message is accompanied of a **timestamp** and a **nonce.**
Before being sent, each message is **signed** with the author's **private key.**

Receiving Messages:

The **freshness** is checked with the **nonce** and **timestamp.**
The message's **signature** is **verified** with the author's **public key**.
The operation's **arguments** are validated.

Operations





Storage

In order to guarantee **durability**, the server's state (goods -> users mapping and nonces) is serialized and stored in non-volatile memory each time it's modified.

This operation is realized atomically - a temp file is created, and only once the serialization has successfully completed, the replace is done with an atomic file move - therefore guaranteeing **atomicity** and **consistency**.

The client's state (list of nonces) is also stored in the same.

Dependability

The server's state is persistently stored, and updates are all performed atomically, so even if the server fails during an update, no data is lost, nor is the server's state left in an invalid state. The same is true for the clients.

Security

The system provides the following security guarantees in it's communication:

- **Integrity**
- **Non-repudiation + Authentication**
- **Freshness**

Non-repudiation and Authentication
Achieved through the use of Digital Signature. All messages exchanged signed with the author's private key, giving a non-repudiable guarantee that he authored that message.
When a message is received claiming to come from a certain user, that user's public Key is used to validate the signature and confirm it was authored by that user.

Integrity
Also provided by the digital signature mechanism. If the original message is tampered with, the verification of the digital signature will fail.

Freshness:
Each and every message is accompanied with a **nonce** and a **timestamp**. When a message is received, the timestamp is verified to be within the limits defined and then the nonce is checked against all the stored nonces received so far. The nonces are persistently stored, so even in case of a server or client shutdown, they will remain persistent.

The use of the timestamp is not strictly necessary to ensure freshness, but it consists in an optimization because, if the timestamp is not within a defined limits, the message is rejected without having to iterate over the stored nonces. It could be further optimized if we deleted the nonces in storage older than that limit.