PolicePatrol_sec5_49582_53587

Client-server communication secure against cyber-attacks with management of the different types of clients that can connect to the server.

Mattia Garofalo
Thomas Charlier

Contents

INTRODUCTION 2
STRUCTURE 2
CLASSES 2
COMMUNICATION 3

- Customer connection 3
- Communication with the server: 3

LIST AND SOLUTION ATTACKS 4
DDOS ATTACK: 4
MAN IN THE MIDDLE: 5
PASSWORD ATTACKS: 7
CLANDESTINE LISTENING: 8
BIRTHDAY ATTACK: 9
PROBLEMS NOT RESOLVED 9
BROCKEN AUTHENTICATION: 9
DATA PERSISTENCE ATTACK: 9
CONCLUSION 10
BIBLIOGRAPHY 11

INTRODUCTION

The goal of the project is to have an exchange of information, between a client and a server, secure to avoid all types of cyber-attacks.

STRUCTURE

The project was created with NetBeans. The classes of the project are written in java.

CLASSES

- BasicMessage.java
- BasicServer.java
- BasicTextClient.java
- Captain.java
- Challenge.java
- ClientMain.java
- Coordinate.java
- MsgTypes.java
- Pair.java
- Patrol.java
- ServerMain.java

COMMUNICATION

- Customer connection

  -The client sends to the server his login (not encode) and his password which is hashed after adding the salt, encoding and sending to the server with a public key, to allow the server to decode and store the hash of the password in the .txt available for the storage of connection information.

- Communication with the server:

  -The server sends a seed (random number) and his public key to the client which requests to connect to the server.

  -The client sends with the message a series of information which allows the server to check if it is receiving a message by an identified client and that the message has not been intercepted and modified.

  -The server sends a random number to encode. (Challenge)

  -The client decodes the random number with his private key by incrementing by 1 before encoding it and sending it to the server.

  After the client, it sends the server a hash, created in relation to the seed received when connecting to the server. (Lamport)

  -The server also does a hash of the seed it sent to the client and it compares the hash it received with the hash it calculated, if the two are equal, the server knows that it is the correct one sender and can therefore read the message that the client sends.

LIST AND SOLUTION ATTACKS

List of computer attacks against which our application must be protected.

We will not talk about sql injections because our project does not contain database processing. The same goes for url injections.
We will not talk about Cross-site scripting attack (The cyber attacker inserts malicious JavaScript into a website's database), because we don’t use sql database and the server is not online so we don't use HTML code.

DDOS ATTACK:

- Distributed Denial of Service attack: it is an attack with the aim the flood of a network in order to crash the server.

  APPLIED SOLUTION:

  - We limit the number of connections to the server (ex 1000) to avoid saturation of the server memory.
  A counter is set up, it is incremented when a connection of a new user and it is decremented on disconnection, to know the number of connections at a time t, reducing DDos attacks as much as possible.

The best solutions for this type of attack are:

- Set up automatic alert systems
- Set up a mirror site
- Distribution of services on different servers (microservices)
- Regular automatic evaluation
- Installation of a firewall or / and an intrusion prevention system (IPS)
- Plan an intelligent network architecture
- Arrange with your internet service provider (ISP)
- Washing machine technology

We've found others solutions but these comport to use external service or programmes for to have information about the traffic connection to the server, like hping3 for Linux and LOIC (low orbit ion cannon) for windows, but we don't use, for this project, those types of solutions.

MAN IN THE MIDDLE:

- Session hijacking: an attacker hijacks a session between a trusted client and a network server. The attacker substitutes the client's IP address while the server continues the session, believing that it is still the client.

- Replay: A replay attack occurs when an attacker intercepts and records old messages and later attempts to send them, pretending to be someone you trust.

SOLUTIONS:

- OTP LAMPORT:
  The Lamport algorithm for generating and applying one-time passwords (OTPs) is a simple solution that provides great value in the right context.
  The core of the Lamport OTP scheme requires that cooperating client / service components agree to use a common sequencing algorithm to generate a set of expiring one-time passwords (client side), and validate client-provided passkeys included in each client-initiated request (service side). The client generates a finite sequence of values starting with a "seed" value, and each successor value is generated by applying some transforming algorithm (or `F(S)` function) to the previous sequence value:

  ```
  S1=Seed, S2=F(S1), S3=F(S2),
  S4=F(S3), ...S[not] =F(S[not-1])
  ```

- CHALLENGE RESPONSE:
  1.when the Server sends a message to Client, he includes an encrypted random integer n1;
  2. when Client send the answers, he provides n1+1 (to be checked by the Server), and includes an encrypted random integer n2;
  3. When the Server send the answers, he provides n2+1 (to be checked by the Client), and includes an encrypted random integer n3, etc.

- KEY SESSION

  All messages are encrypted with a session key, which is also transmitted securely using asymmetric encryption.

  The main advantage of using this kind of session key instead of using the private and public keys is that the exchange between the client and the server is faster.

PASSWORD ATTACKS:

ATTACK:

- By brute force: guessing a password by entering what people enter most often: last name, first name, favourite hobby, children's dates of birth, etc. For this attack, the hacker sends a very large number of requests to test the possibilities of passwords until they find the right one.

SOLUTION:

- To protect against this attack, we limit the number of connection attempts to 3 errors. After the 3rd error, authentication is disabled for this user for 5 minutes.

ATTACK:

- By dictionary: this consists of copying an encrypted file containing common passwords and comparing the results.

SOLUTION:

- To protect against this type of attack, the user is asked to use a password of at least 8 characters and one or more numeric character. This way a user's password will be more robust.

ADDITIONAL SOLUTION

- Creation of salt, which is a sequence of random characters, which will be appended to the end of the password given by the user before being hashed and stored.

  Impossible for people who try an attack to have the same salt as the salt created by the server and it is also impossible to know the number of hash recurrence for the password, before storage.

CLANDESTINE LISTENING:

Eavesdropping or eavesdropping is the result of interception of network traffic.

ATTACKS:

- Passive eavesdropping: a hacker intercepts data by listening to the transmission of messages on the network.

- Active eavesdropping: A hacker actively seizes information by posing as a friendly unit and sending requests to transmitters.

SOLUTION:

The data are encrypted.

BIRTHDAY ATTACK:

The birthday attack is a type of cryptographic attack, which exploits the mathematics behind the birthday problem in probability theory. The hackers uses this probabilistic model to reduce the complexity of cracking a hash function.

SOLUTION:

For this project we don't use a hash for the signature so this type of attack can't have consequences.

PROBLEMS NOT RESOLVED:

BROCKEN AUTHENTICATION:

We don’t find how to put a timeout for delete an id user from the list of the users online, if the user quit the session without a correct disconnection.

So, the project is vulnerable for the attacks using this type of failure to use a user id for to communicate with the server.

DATA PERSISTENCE ATTACK:

- In this type of attack, the hacker tries to access sensitive data that is stored in program variables. For the purpose of modifying this data or recovering it.
- Since our program is written in java, the additional layer of variable processing provided by the java-machine makes data persistence attacks more difficult. But in other way the garbage collector makes copies on RAM of variables so we can’t really secure erase the data.

CONCLUSION

The application create for this project is secure to avoid most types of cyber-attacks but not the all types.

We are found a lot of difficulty to find the way how to secure, the communication between the server and the client, for some type of cyber-attacks, but we are satisfied for our application and we are improved our knowledge about cyber-security

We know is possible to make the application more secure, with some external implementation, but we do the possible for this project.

BIBLIOGRAPHY

LAMPORT:
https://www.infoworld.com/article/2078022/lamport-s-one-time-password-algorithm--or--don-t-talk-to-complete-strangers--.html

BIRTHDAY ATTACK:
https://www.sciencedirect.com/topics/computer-science/birthday-attack