

## Goal

The objective of this session is to get a first introduction to the sockets, a mechanism of communication between processes that may be running on the same or different systems.

## Motivation

More specifically, with this session, the student has to exercise:

- Creation of sockets (socket)
- Establishment of connections (connect)
- Connection closure (close)
- Sending messages between client and server (read, write)
- Passing parameters to an executable (argc, argv)

## Previous documentation

To complete this session, it is recommended that you read the following references:

SALVADOR J., CANALETA X. (2014). *Programació en UNIX per a pràctiques de Sistemes Operatius*, Publicacions d'Enginyeria i Arquitectura La Salle (Edició PDF). Capítulo 7. Pág. 97 en adelante.

## Guardian of Enigmas

In the vast ocean of data in the digital world, a legendary treasure known as "The Code Chest" has been hidden on a mysterious island protected by the server Guardian of Enigmas. This server challenges all navigators (clients) who seek to uncover the encoded secrets within the chest. Each challenge they overcome brings them closer to the treasure, revealing coordinates piece by piece.

Your goal is to build a server in C that functions as the guardian of the chest, challenging clients with cultural, logical, and mathematical riddles. The server must handle multiple connections simultaneously, provide challenges, validate responses, offer hints, and track the progress of each client separately. This server must be able to:

- **Initiate and Manage Connections:** Listen on the port and accept multiple incoming connections.
- **Load Challenges:** Read a set of challenges from a text file (challenges.txt) upon startup and keep them in memory for users to check.
- **Dynamic Interaction:** Respond to challenge requests, receive responses, provide hints, and display progress.

Keep in mind the client you already implemented the client, so you understand the type of requests you'll be receiving. These were:

1. **Receive Current Challenge:** The server will receive a "1\n" and the it must send the corresponding challenge to the client extracted from the file.
2. **Send Response to Challenge:** The server will receive a "2\n" followed by "<answer>\n". The server must verify the response to the corresponding challenge extracted from the file.
3. **Request Hint:** The server will receive a "3\n" and must send a hint for the challenge corresponding to the client extracted from the file.
4. **View Current Mission Status:** The server will receive a "4\n" and must then inform how many challenges remain, and if all are completed, provide the coordinates of the chest. (This includes tracking each client: which challenge they are on and how many they have left to finish).
5. **Terminate Connection and Exit:** The server will receive a "5\n" and must handle the client's request to disconnect.

**Structure of the "challenges.txt" file:**

Question1|Answer1&Hint\n

Question2|Answer2&Hint\n

Question3|Answer3&Hint\n

....

```

=====
||      Navigation Menu      ||
=====
1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit
Select an option: 2
Enter your response to the challenge: 100
Incorrect answer, try again.

```

## Example 1 - Challenge

Server:

```
  \____; ;____\
  | /          \
  |  ( )oo( )  |
  | \(%()*^^( )^ \
  |  |-%-----|
  |  \ | %  ))  |
  |  \ | %-----|
```

Welcome to the Guardian of Enigmas Server. Prepare to embark on a journey of puzzles and mysteries!

Welcome Cris!

Cris - request challenge...

Cris - sending answer...

Checking answer...

□

Client:

```
=====
||      Navigation Menu      ||
=====
```

1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit

Select an option: 1

Challenge: Result of 123 + 234?

```
=====
||      Navigation Menu      ||
=====
```

1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit

Select an option: 2

Enter your response to the challenge:

Correct answer! Proceeding to the next challenge.

**Example 2 – View current mission Status and Request Hint**

Server:

```
  \____;_____\
  | /          \
  |  ( )oo( )  |
  | \(%()*^^( )^ \
  |  |_____|  |
  |  | %  )  |
  |  | %  |  |
  \  | %  |  |
```

```
Welcome to the Guardian of Enigmas Server. Prepare to embark on a journey of puzzles and mysteries!
Welcome Cris!
Cris - request challenge...
Cris - sending answer...
Checking answer...
Cris - request to view current mission status...
Cris - request challenge...
Cris - request hint...
Hint sent!
□
```

Client:

```
=====
||      Navigation Menu      ||
=====
1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit
Select an option: 4
Current mission status:
You are currently on challenge 2 out of 5.

=====
||      Navigation Menu      ||
=====
1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit
Select an option: 1
Challenge: Decode this text: "Khoor Zruog"

=====
||      Navigation Menu      ||
=====
1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit
Select an option: 3
Your hint is coming...
Use Caesar cipher with a shift of -1.
```

Current mission status:  
Congratulations! You've found the treasure at coordinates: X:100, Y:200. Disconnecting.

Terminate connections:

Server:

```

      _____\
     /            \
    /              \
   /                \
  /                  \
 /                    \
/                      \
\                      /
 \                    /
  \                  /
   \                /
    \              /
     \            /
      \          /
       \        /
        \      /
         \    /
          \  /
           \/

```

Welcome to the Guardian of Enigmas Server. Prepare to embark on a journey of puzzles and mysteries!

Welcome Cris!

Cris - request challenge...

Cris - sending answer...

Checking answer...

Cris - decide to terminate the connection

^CExiting server...

Client:

```

=====
||      Navigation Menu      ||
=====
1. Receive Current Challenge
2. Send Response to Challenge
3. Request Hint
4. View Current Mission Status
5. Terminate Connection and Exit
Select an option: 5
Terminating connection...

```



## Considerations

- The IP and port to which the client must connect and the server must listen are passed as a parameter.
- It can be assumed that the format of the input parameters will always be correct. However, must check if the number of parameters is correct.
- The port assigned to the client and server is the one assigned to the group.
- The client is guaranteed to always enter an input followed by a return (\n).
- The server manages who is the winner, clients must only save and display the board distribution.
- The communication between client and server must be carried out following the guidelines of the frames indicated in the statement.
- The output of the program must be similar to that of the statement (see execution examples).
- The use of “system” or “popen” or analogous functions of the same family is not allowed.
- The use of global variables must be reduced to the minimum possible to make the program work correctly.
- All input and output must be done with file descriptors, the use of printf, scanf, FILE\*, getchar, or similar is not allowed.
- It must be compiled using the -Wall, -Wextra and -lpthread flags.
- Any practice that contains warnings will be directly discarded.
- All resources must be released.
- A single “S5.c” file must be delivered that will have the names and logins of the group members commented. Otherwise, the practice will not be corrected.
- You may use the client you made last session. However, the correction will be made with the one provided / last session solution.