

# Practical Work 2: RPC File Transfer

Vo Truong Giang - BI12-130

November 29, 2025

## 1 Introduction

In this practical work, I upgraded the file transfer system from TCP sockets to Remote Procedure Calls (RPC) using the ONC RPC (Sun RPC) standard.

## 2 RPC Service Design

Unlike the streaming approach of TCP sockets, RPC allows the client to invoke a function on the server directly.

- **Input:** A structure containing the filename (string) and file content (opaque data).
- **Output:** An integer status code (1 for success, 0 for failure).
- **Tool used:** `rpcgen` to generate stubs and XDR filters.

## 3 System Organization

The system consists of:

- **transfer.x:** The Interface Definition Language (IDL) file.
- **Server:** Implements the `upload_file_1_svc` function to write data to disk.
- **Client:** Reads the file into memory and calls `upload_file_1`.

## 4 Implementation

### 4.1 Protocol Definition (.x file)

```
struct file_data {
    string name<256>;
    opaque data<>;
    int bytes_sent;
};

program FILE_TRANSFER_PROG {
    version V1 {
        int UPLOAD_FILE(file_data) = 1;
    } = 1;
} = 0x31230000;
```

## 5 Conclusion

The RPC implementation abstracts the networking details, allowing the developer to focus on the logic of function calls. The system successfully transfers files using the defined RPC interface.