experiment/payload.cpp

```cpp
/**
 * Payload Generator
 * Author: Arnab Ghosh
 * Date: 11/3/2023
 *
 * Payload generator generates a random character string file of a certain size.
 * Payload generator shall be used during the method in order to create the given payload.
 *
 * Usage: payload sample_file_name 16000
 * Note that the size of the file is specified in megabytes.
*/
#include <iostream>
#include <fstream>
#include <string>
#include <random>

using std::string;

/**
 * Character array of all alphanumeric characters by which the random string will be constructed.
*/
const char CHARSET[] = "abcdefghijklmnopqrstuvwxyzABCDEFHIJKLMNOPQRSTUVWXYZ1234567890";
const int CHARSET_LENGTH = 62; // 26 lowercase + 26 uppercase + 10 digits

/**
 * Returns random character from default charset.
*/
char getRandomChar() {
    int index = std::rand() % CHARSET_LENGTH;
    return CHARSET[index];
}

/**
 * Prints help menu, describing how to use program. Printed when incorrect number of arguments
presented.
*/
void printHelpMenu(const string& message) {
    using std::cout;
    cout << message << "\n";
    cout << "Usage: generate a payload file with random characters of a given size" << "\n";
    cout << "Example: payload sample_file_name 16000" << "\n";
}

/**
 * Entry point for program. Main program logic occurs here.
*/
int main(int argc, char* argv[]) {
    // Set up parameters. This assumes the strict pattern followed as above.
    string filename = argv[1];
    int filesize = std::stoi(argv[2]);

    // testing
    std::cout << filename << " " << filesize << std::endl;
```

```
53
54      // create file and fill with random bytes
55      std::ofstream buffer(filename);
56      for(int i = 0; i < 1000000 /* This is 1 Million; */; i++) {
57          for(int j = 0; j < filesize; j++) {
58              buffer << getRandomChar();
59          }
60      }
61
62      buffer.close();
63
64
65      return 0;
66  }
```