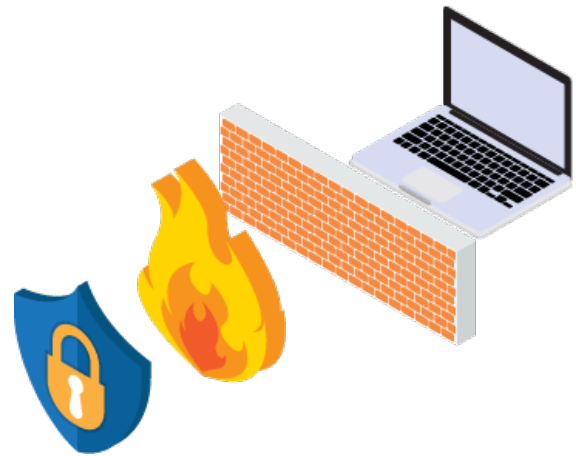




# Security Device Application

Let us create a program that will check whether an incoming IP address is allowed or blocked based on a list of permitted IP addresses. This C++ example will help you understand how firewall rules can be simulated in a controlled environment, providing a foundational understanding of how firewalls operate at a basic level.



**Alt text:** *Developing a firewall*

Here is a simulated Firewall Rule in C++:

## Step 1: Include Necessary Libraries

```
#include <iostream>    // For input/output operations
#include <string>       // For handling strings
#include <vector>       // For using the vector container
```

- This step imports the necessary standard libraries for input/output operations, string handling, and the vector container.

## Step 2: Function to Check if the IP is Allowed

```
bool isAllowedIP(const std::string& ip, const
std::vector<std::string>& allowedIPs) {
    for (const auto& allowedIP : allowedIPs) {
        if (ip == allowedIP) {
            return true; // Allow traffic from this IP
        }
    }
    return false; // Block traffic from this IP
}
```

- This function `isAllowedIP()` accepts an IP address (`ip`) and a list of allowed IP addresses (`allowedIPs`).
- It loops through the allowed IP addresses. If the incoming IP matches one of the allowed IPs, it returns true to allow the traffic.
- If no match is found, it returns false, indicating that traffic should be blocked.

### Step 3: Initialise List of Allowed IPs and Capture Input

```
int main() { std::vector<std::string> allowedIPs =  
{"192.168.1.100", "192.168.1.101"};  
  
// Example allowed IPs std::string incomingIP; std::cout <<  
"Enter the incoming IP address: "; std::cin >> incomingIP;
```

- In the `main()` function, a vector of allowed IP addresses is initialised.
- The program prompts the user to enter an incoming IP address, which is stored in the `incomingIP` variable.

### Step 4: Check if the IP is Allowed and Print Result

```
if (isAllowedIP(incomingIP, allowedIPs)) {  
    std::cout << "Access Granted: Traffic from " <<  
incomingIP << " is allowed." << std::endl;  
} else {  
    std::cout << "Access Denied: Traffic from " <<  
incomingIP << " is blocked." << std::endl;  
}
```

- The `isAllowedIP()` function is called with the user's input (`incomingIP`) and the list of allowed IPs.
- If the IP is allowed, the program outputs "Access Granted".
- Otherwise, it outputs "Access Denied".

## Step 5: End the Program

```
return 0;
}
```

- The program ends with return 0;, indicating successful execution.

## Explanation of Terms

**Allowed IPs:** A list of IP addresses that are allowed to access the system.

**Is Allowed IP:** A function that checks if the incoming IP address is in the list of allowed IPs.

**User Input:** The program asks for an incoming IP address and checks if it is allowed or blocked.



**Alt text:** Understanding a code

## How It Works:

The user inputs an IP address. The program checks if the IP address is in the allowed list.

- If it is, the program prints "*Access Granted*"
- Otherwise, it prints "*Access Denied*"