CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY

Department of Electronics and Communication Engineering

Gandipet, Hyderabad – 500075

PROJECT REPORT

ON

CLIENT/SERVER SECURITY SYSTEM USING LABVIEW

Submitted by (ECE 2)

A. APURVA	**(160118735063)**
ASHWITHA K	**(160118735064)**
M.SANJANA	**(160118735079)**

TABLE OF CONTENTS

CLIENT/SERVER SECURITY SYSTEM USING LABVIEW

ABSTRACT

Any security system's name contains the most fundamental definition. It is essentially a method or means of securing something through a system of interconnected components and devices. They are electronic device networks that function with a central control panel to protect against potential intruders.

The security system provides for one indicator light for each zone. The colour of the light provides the status information of the zone. The colours of the zone are:

- Green: Normal
- Red: Alarm
- Blue: Bypass
- Orange: Tamper

This system is a Client/Server application. The Client side should transmit signals to the server side via a TCP/IP connection. A Bypass input should always override an Alarm input but should not turnoff an existing Alarm condition. An Alarm condition should not be included while a zone in a Bypass condition. A Tamper input should always override both an Alarm input and/or a Bypass input but should not turn off existing Alarm and/or a Bypass condition. Alarm and Bypass conditions should not be indicated while a zone is in a Tamper condition. An existing condition should not be indicated when an overriding input is removed. The LabVIEW programme is more ideal for this process because of its ease of operation and use.

1. INTRODUCTION

Security camera systems have gained popularity in today's modern world. For any number of reasons, the person who administers your system may have to meet a certain level of security. For instance, the security level might be a matter of corporate policy. These security standards might be applied to the network, the operating system, application software, even programs written by the person who administers your system.

This project describes the security features provided with Transmission Control Protocol/Internet Protocol (TCP/IP), both in standard mode and as a secure system, and discusses some security considerations that are appropriate in a network environment. Many of the security features available for TCP/IP are based on those available through the operating system.

2. AIM

To implement a client and server VI using TCP protocol for security zones indication using LabVIEW. Also to send date and time to an excel sheet if an alarm, bypass, or tampering of the system occurs.

3. OBJECTIVE

- To develop a client vi and server vi that can communicate with one another in the event of a security system alarm, bypass, or tamper.
- To show a change in the colour of the boolean led on the server side when a condition changes on the client side.
- To save the data regarding changes in the condition of the security system at a given date and time to a excel spreadsheet.

4. DESCRIPTION

4.1 TCP/IP

TCP/IP stands for Transmission Control Protocol/Internet Protocol and is a suite of communication protocols used to interconnect network devices on the internet. TCP/IP is also used as a communications protocol in a private computer network.TCP/IP specifies how data is exchanged over the internet by providing end-to-end communications that identify how it should be broken into packets, addressed, transmitted, routed and received at the destination. TCP/IP requires little central management and is designed to make networks reliable with the ability to recover automatically from the failure of any device on the network.

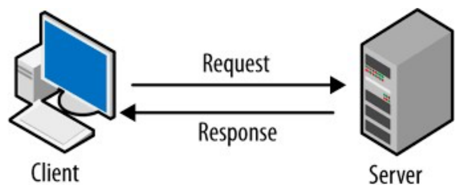
TCP/IP can be used to provide remote login over the network for interactive file transfer to deliver email, to deliver webpages over the network and to remotely access a server host's file system. Most broadly, it is used to represent how information changes form as it travels over a network from the concrete physical layer to the abstract application layer. It details the basic protocols, or methods of communication, at each layer as information passes through.

4.2 CLIENT/SERVER ARCHITECTURE

The Client-server model is a distributed application structure that partitions task or workload between the providers of a resource or service, called servers, and service requesters called clients. In the client-server architecture, when the client computer sends a request for data to the server through the internet, the server accepts the requested process and deliver the data packets requested back to the client. Clients do not share any of their resources.

Client: When we talk the word Client, it means to talk of a person or an organization using a particular service. Similarly in the digital world a client is a computer (Host) i.e., capable of receiving information or using a particular service from the service providers (Servers).

Servers: Similarly, when we talk the word Servers, it means a person or medium that serves something. Similarly in this digital world a Server is a remote computer which provides information (data) or access to services.



4.3 LABVIEW

Laboratory Virtual Instrument Engineering Workbench (LabVIEW) is a system-design platform and development environment for a visual programming language from National Instruments.

LabVIEW integrates the creation of user interfaces (termed front panels) into the development cycle. LabVIEW programs-subroutines are termed virtual instruments (VIs). Each VI has three components: a block diagram, a front panel, and a connector panel. The last is used to represent the VI in the block diagrams of other, calling VIs. The front panel is built using controls and indicators. Controls are inputs: they allow a user to supply information to the VI. Indicators are outputs: they indicate, or display, the results based on the inputs given to the VI. The back panel, which is a block diagram, contains the graphical source code. Allthe objects placed on the front panel will appear on the back panel as terminals. The back panel also contains structures and functions which perform operations on controls and supply data to indicators. The structures and functions are found on the Functions palette and can be placed on the back panel. Collectively controls, indicators, structures, and functions are referred to as nodes. Nodes are connected to one another using wires, e.g., two controls and an indicator can be wired to the addition function so that the indicator displays the sum of the two controls. Thus, a virtual instrument can be run as either a program, with the front panel serving as a user interface, or, when dropped as a node onto the block diagram, the front panel defines the inputs and outputs for the node through the connector panel. This implies each VI can be easily tested before being embedded as a subroutine into a largerprogram.

The graphical approach also allows nonprogrammers to build programs by dragging and dropping virtual representations of lab equipment with which they are already familiar. The LabVIEW programming environment, with the included examples and documentation, makes it simple to create small applications. This is a benefit on one side, but there is also a certain danger of underestimating the expertise needed for high-quality G programming. For complex algorithms or large-scale code, it is important that a programmer possess an extensive knowledge of the special LabVIEW syntax and the topology of its memory management. The most advanced LabVIEW development systems offer the ability to build stand-alone applications. Furthermore, it is possible to create distributed applications, which communicate by a client– server model, and are thus easier to implement due to the inherently parallel nature of G.

5. Methodology

To create client VI

- Open new VI using Ctrl+N and Ctrl+T.
- In front panel select –- >Silver--> Boolean -->Push button. Repeat it three times and name them as Alarm, Bypass and Tamper.
- Go to Block diagram --> Structures-->While loop. Create control for the stop condition in while loop.
- Data communication-->Protocols-->TCP --> TCP open connection and place it outside of the while loop.
- Data communication--> Protocols--> TCP-->TCP close connection and place it outside of the while loop.
- Data communication-->Protocols-->TCP-->TCP write and place it within the while loop.
- Right click-->Comparison--> Select. Connect the select block to boolean push button block.
- Select two numeric indicators and change the numbers to 0 and 1. Connect 0 to the false condition of select and 1 to the true condition of select.
- Repeat the above two steps 3 times. For bypass and Tamper conditions change the numeric constant value on connected to true of select to 2 and 4 respectively.
- Select compound arithmetic and connect the 3 select block outputs as inputs.
- Go to structures-->case structure. Connect output of compound arithmetic to casestructure.
- Add a total of 8 cases in the case structure and inside each case include a string constant with the case number and another string constant stating the condition (Normal, Alarm, Bypass or Tamper). For case 0 the condition is normal. For case 1 it is alarm. For cases 2 and 3 it is bypass and for remaining all cases it is tamper condition.
- Connect the String constant containing number in each case to the data in of TCP read block.
- Initialize remote port and timeout in TCP open connection block. The timeout value should be made as -1 so that the execution does not timeout until the connection between client and server is established.
- Right click error out of TCP close connection--> dialog and user interface-->simple error.
- Select timing-->wait. Set constant to 100ms.
- For sending the data to the excel sheet right click on block diagram-->file I/O-->Write delimited spreadsheet.
- Create file path control for write delimited spreadsheet block.
- Go to timing-->get date/time and get date/time in seconds.
- Go to array-->build array to initialize a one day array thetakes date, time, and condition.

- Connect output of build array block to delimited spreadsheet block.

To create server VI

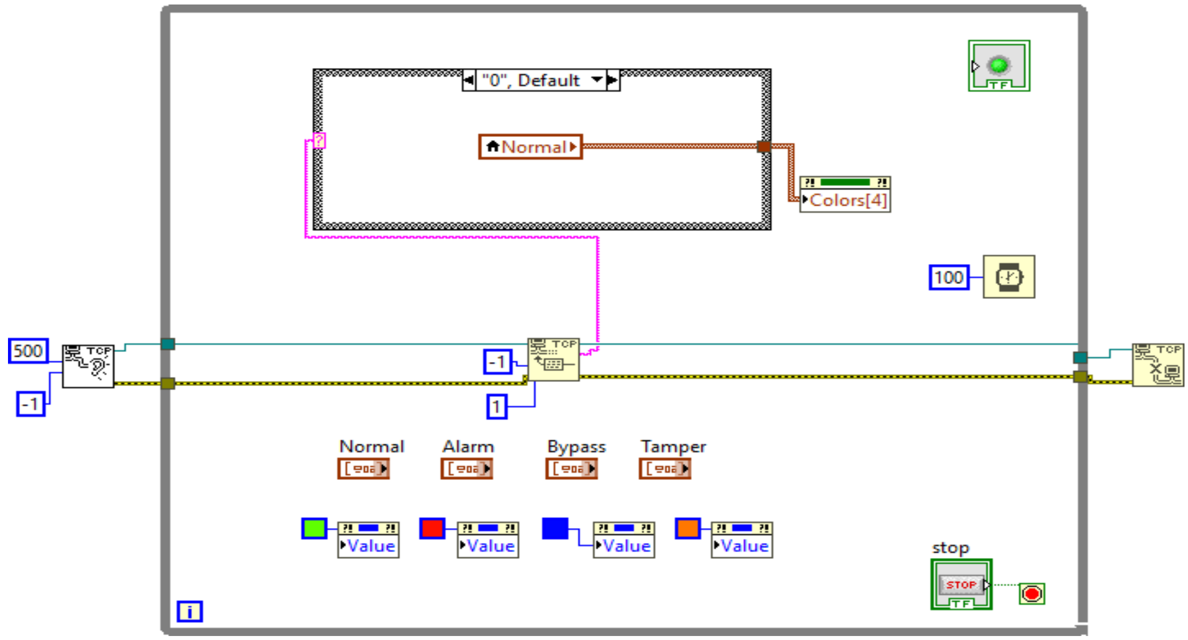
- Go to front panel--> Select Boolean-->Round LED.
- Go to Block Diagram -->Right click--> select structures-->While loop and create stop condition.
- Data communication--> Protocols-->TCP-->TCP listen and place it outside of the while loop.
- Data communication--> Protocols-->TCP-->TCP close connection and place it outside of the while loop.
- Data communication-->Protocols-->TCP-->TCP read and place it inside of the while loop.
- For TCP read block create constant at number of packet bytes to read and change its value to 1.
- Keep timeout value for TCP read block as -1.
- Initialize remote port number and timeout condition for TCP listen.
- Right click on block diagram-->select structures-->case structure. Convert Boolean case into string case structure. Add 8 cases.
- Connect data out of TCP read block to case Boolean.
- Create property node colors for the Boolean indicator, change to write and create control. Repeat 4 times. (Each control represents the condition normal, alarm, bypass and tamper).
- Create a local variable for each of the controls and place them in the corresponding case of the case structure. Connect the local variable for each case to the property node.
- In the front panel, create property node value for each of the color indicator. Change all to write. Connect color box constant to the property node. Repeat 4 times. By doing this we do not have to reinitialize each time we close and reopen the vi.

To run the VI

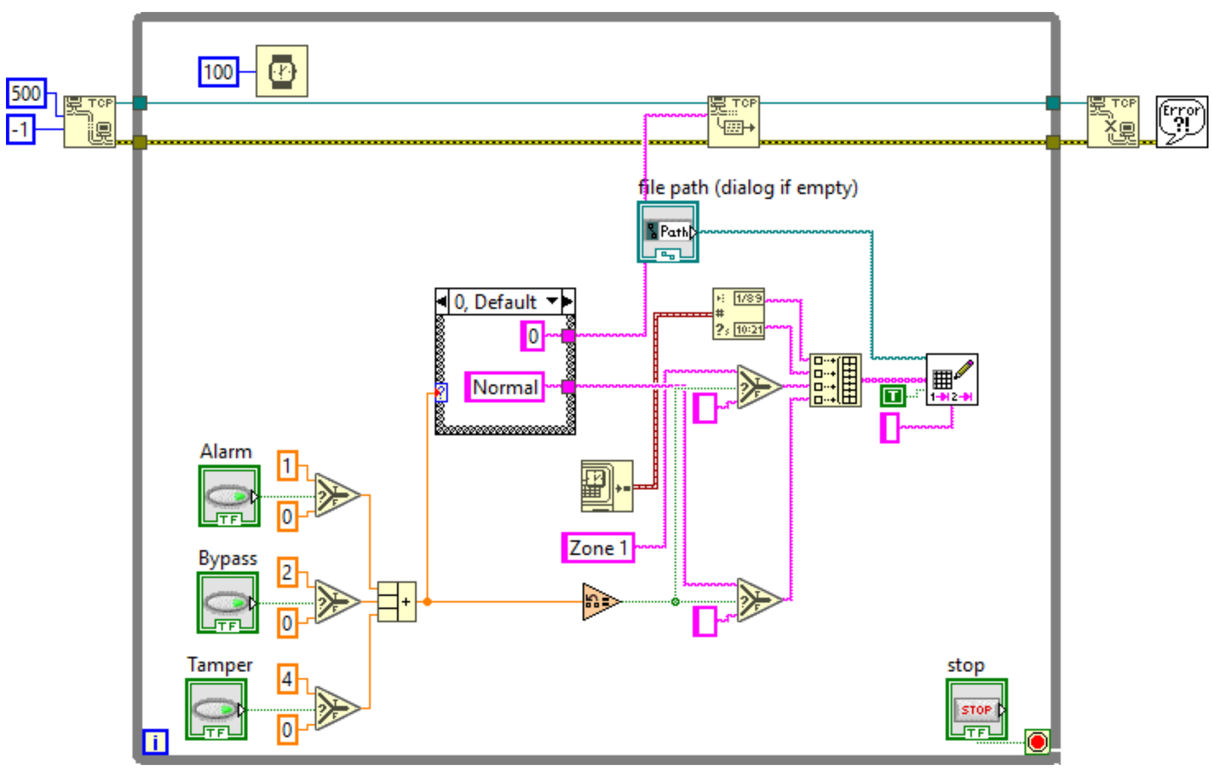
- Create an empty excel file in the format .csv at some location on the PC.
- Arrange the front panels of both the client vi and server vi side by side.
- Run both the Vis.
- Click on the Boolean push buttons in the client front panel. When none of the buttons are entered, the led will be green. If only alarm is entered led will turn red. If bypass I entered led will turn blue even though alarm is entered. If tamper is clicked the led will turn orange color even though the other push buttons are entered.
- Stop the execution and open the excel sheet. The date, time and condition at the time will have been updated into the excel sheet.

6. SIMULATION

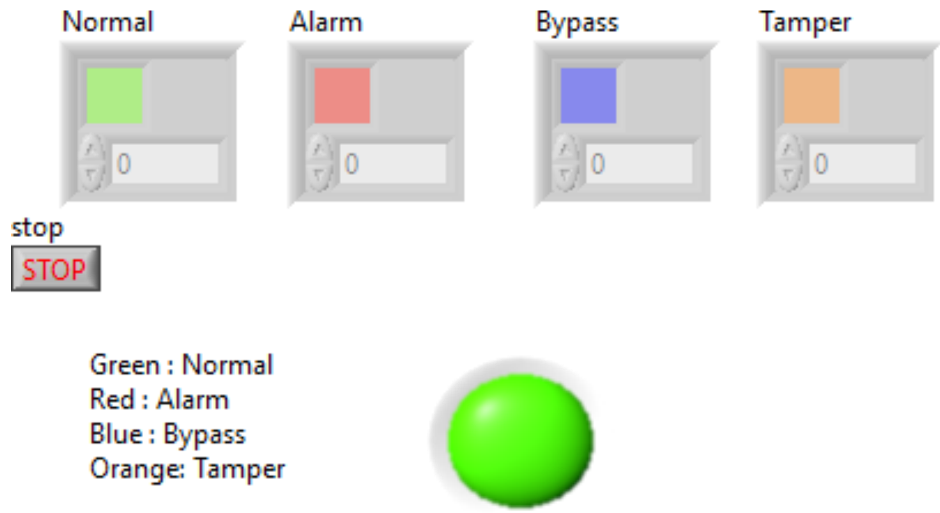
6.1 SERVER BLOCKDIAGRAM



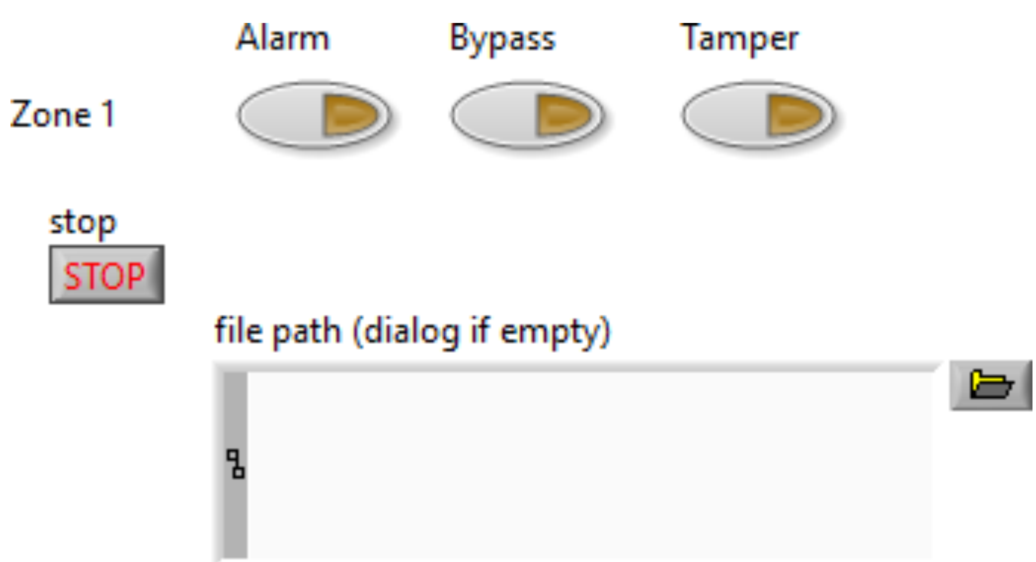
6.2 CLIENT BLOCKDIAGRAM



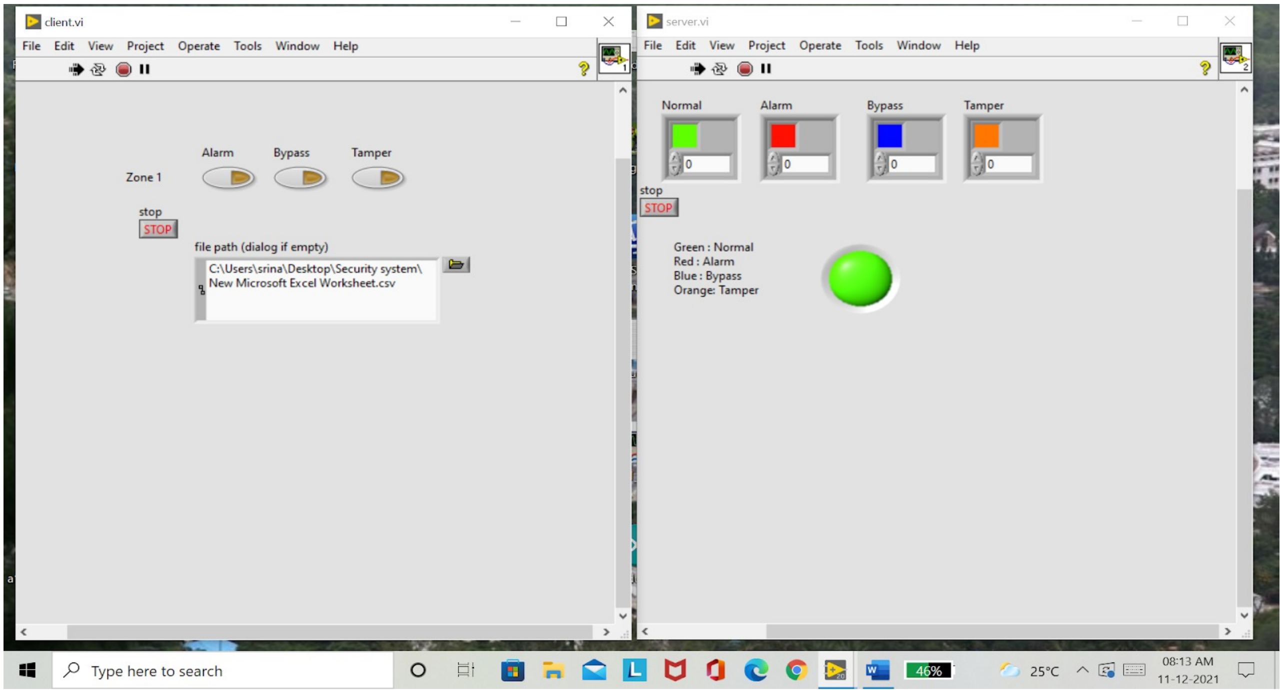
6.3 SERVER FRONT PANEL



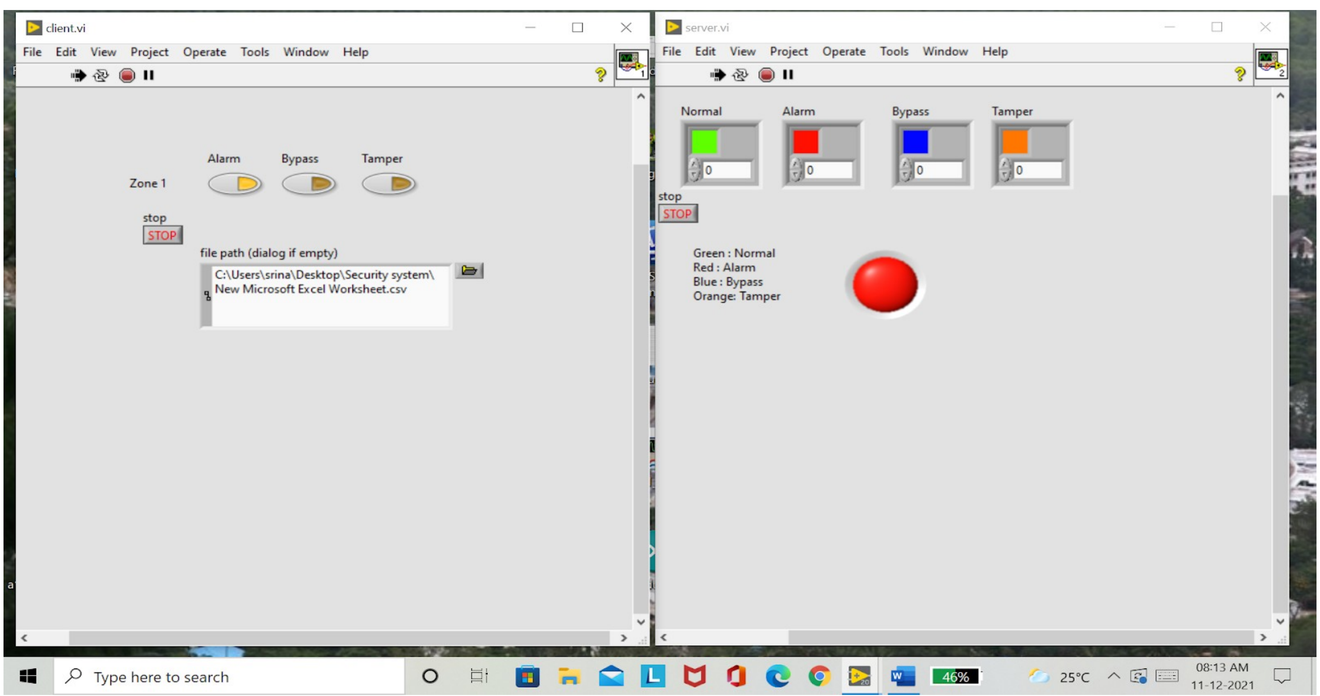
6.4 CLIENT FRONT PANEL



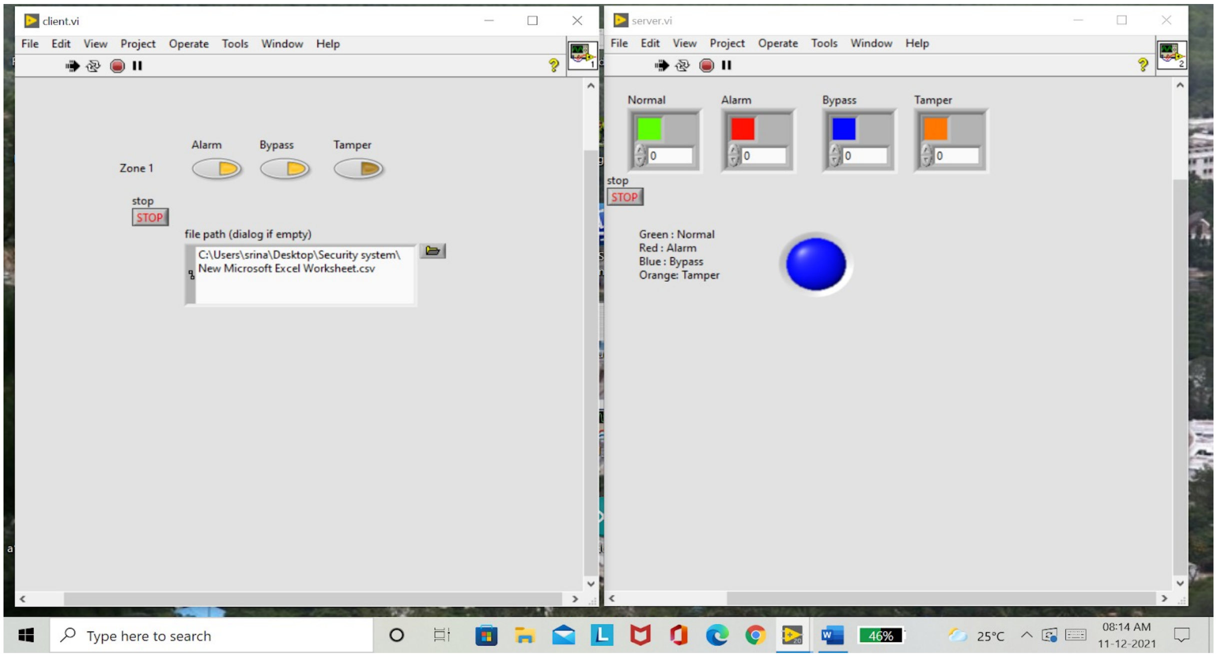
7. RESULTS:



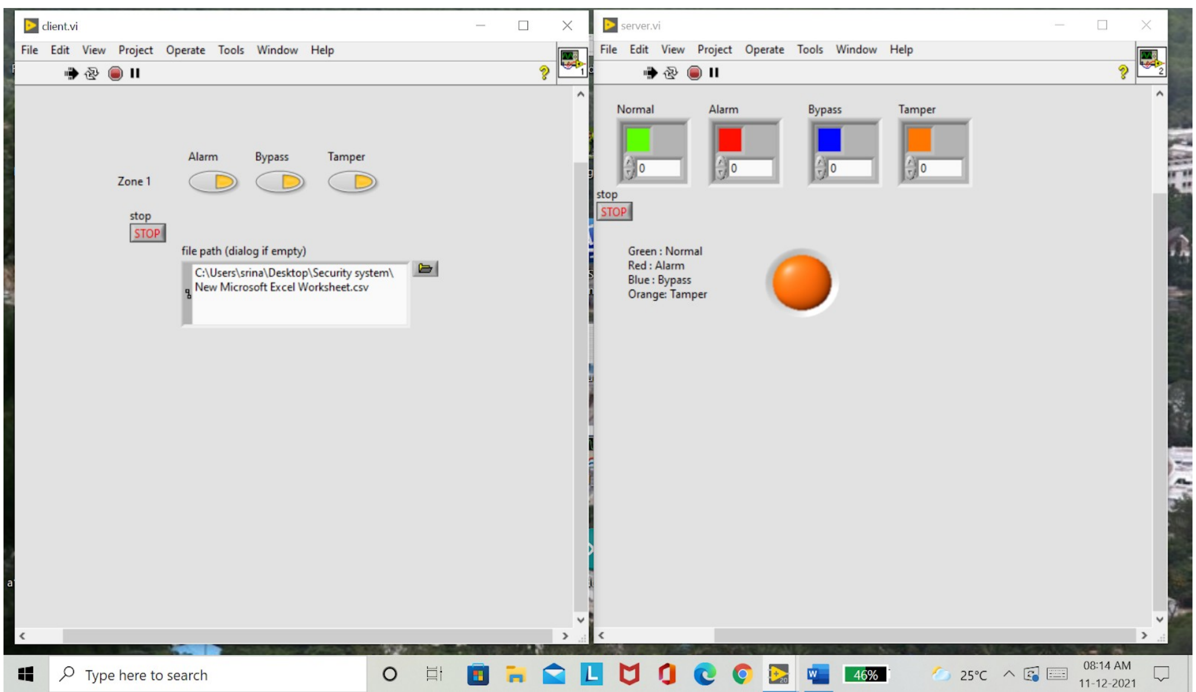
Fig(1) shows the normal condition



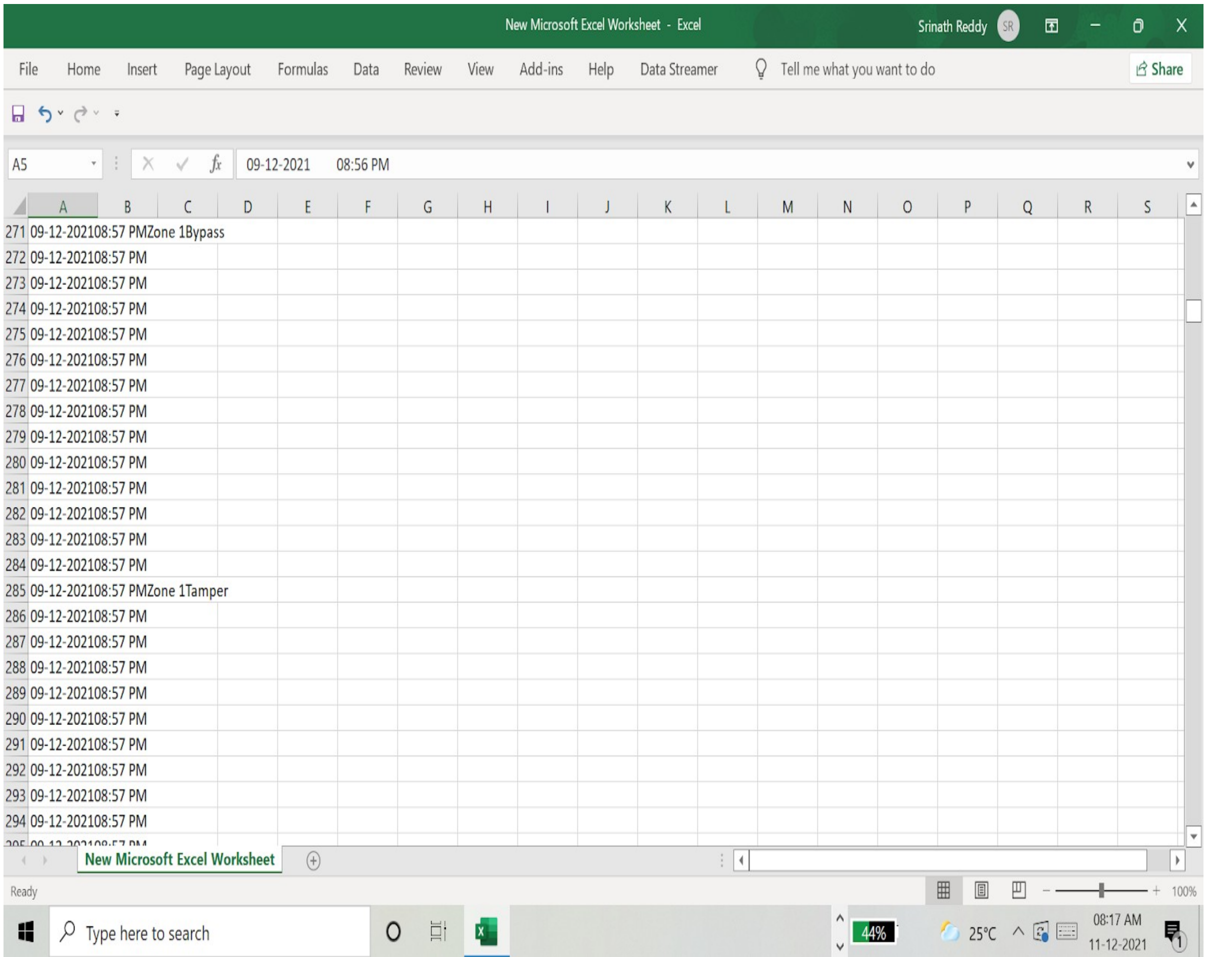
Fig(2) shows the alarm state



Fig(3) shows the bypass state



Fig(4) shows the tamper state



Fig(5) shows the data collected in the excel sheet

8. DISCUSSIONS

The advantages of a client-server network are greater security of the network, more control e.g. network traffic passing through the network, being able to see what each computer is doing and limiting certain actions and preventing things such as viruses spreading, also the amount of data storage available to each computer on the network.
TCP/IP Communication provides a simple user interface that conceals the complexities of ensuring reliable network communications.

9. CONCLUSIONS

A client/server architecture for a security system has been simulated using LabVIEW. The client and server vis are connected with a TCP connection. By changing the Boolean push buttons on the front panel of the client vi the led indicator in the server vi changes its color.
The date and time at which the conditions of the security system are changed is updated to an excel spreadsheet. Therefore, the security system can be remotely monitored.

10. REFERENCES

[1] RahmatSanudin, Member, IEEE,"Small-scale Monitoring System on LabVIEW Platform"

Available:https://www.researchgate.net/publication/224584707_Small-scale_monitoring_system_on_LabVIEW_platform

[2] Huadong Li; Yufang Zhong; Mingguang Wu " Research on network of remote real-time surveillance system based on LabVIEW", 7th IEEE International Conference on Industrial Informatics. 23-26 June 2009.

[3] Basil Ahmed, "Real-Time Application Surveillance Security System Based on LabVIEW," International Journal of Computer Theory and Engineering vol3.

https://www.researchgate.net/publication/257138986_Real-Time_Application_Surveillance_Security_System_Based_on_LabVIEW