TEST Engineering & Management

May – June 2020
ISSN: 0193-4120 Page No. 2362 - 2365

On Enhancing the Communication with Embedded Systems through Implementation of E-mail Extension Server

[1]M Kranthi Kumar, [2]E T Praveen, [3]J Thrisul Kumar, [4]N Durga Rao
[1,2,3,4]Asst.Professor, Dept of ECE,Vignan'sNirula Institute of Technology and Science for Women AndhraPradesh, India.

Article Info
Volume 83
Page Number: 2362 - 2365
Publication Issue:
May - June 2020

Abstract:

Communicating between the things has become the biggest endeavour. Establishing a smart communication with Internet as a backbone has become a vogue/gravitates these days. To achieve this connection of embedded systems that monitor and control any type of physical equipment to Internet is needed. For example- controlling of the home appliances, safety critical systems. Every embedded board has TCP/IP communication support the code of which is made available as a part of native ROM. Some of the IDEs also are making available TCP/IP software as library, thus making it possible to communicate between an embedded board and any computing location connected on to an internet. The controlling data at times is sent from remote locations using internet which can be sent as an email message, a data packer, and SMS message to the embedded system. To establish such communication an E-mail Extension server must be implemented. Since an E-mail Extension server requires huge space and storage capacity which is not the part of an embedded system gave the necessity for developing core, sleek and meek E-mail server-the miniaturized version on to the embedded microcontroller systems leading to seamless communication between the embedded system and a remote computing location

Article History
Article Received*: 11August 2019*
Revised*: 18November 2019*
Accepted*: 23January 2020*
Publication*:10May2020*

Keywords:*embedded systems, Email server, TCP/IP, miniaturized version*

I. INTRODUCTION

Communication plays a lead role in our life and it may be through any way like file exchange, data exchange and email exchange etc. From short Ideas to lengthy messages we can enable to send by using it.

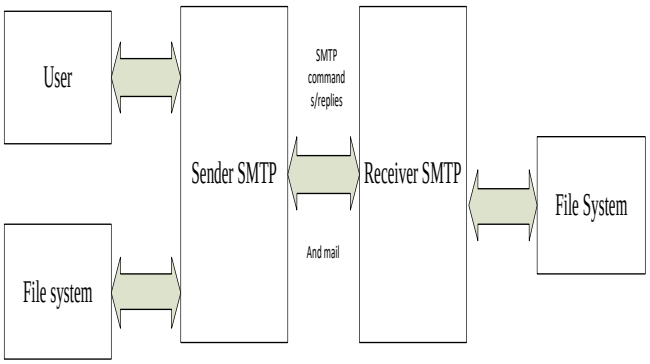
Email system has numerous conventions, each of which has a particular activity during the time spent sending or receiving E-mail messages the generally utilized E-mail structure protocols incorporate SMTP for sending and POP3 protocol for recovering E-mail messages. Upon the establishment of TCP association among the client along with the server, the client sends the email message to the server by utilizing the SMTP protocol. With the similar protocol, the server begins directing the email message to another email server, or sends it to the beneficiary's regional E-mail server.

SMTP is basic connection-oriented, content based ASCII protocol. The sending machine which is working as the client, hold for the receiving machine which is working as a server, to talk first. The server begins by sending a line of content giving its existence and advising whether it is set up to get mail. If not, the client discharges the association and tries again later. In the event that the server will acknowledge email, the client declares from whom the email is coming and to whom it is going. In the event that such a beneficiary exists at the goal, the server gives the customer the approval order to send the message. At that point the customer sends the message and the server recognize it. No checksums are required in light of the fact that TCP gives a solid

May – June 2020
ISSN: 0193-4120 Page No. 2362 - 2365

byte stream. On the off chance that there is more email, that is sent and all the email has been traded in both bearings, the connection isdischarged. The block diagram for SMTP is shown in Figure 1.

The Post Office Protocol (POP3) is an Internet standard protocol utilized by local email programming clients to receive messages from a remote mail server over a TCP/IP protocol.Email servers facilitated by Internet service co-ops additionally utilize POP3 to get and hold messages proposed for their supporters. Occasionally, these subscribers will utilize email client programming to check their inbox of mail on the remote server and download any messages routed to them.Once the email customer has downloaded the messages, they are normally erased from the server, albeit some email customers enable clients to indicate that sends be duplicated or saved on the server for a timeframe.



II. Problem Definition

Implementing email server requires huge amount of resources in terms of memory, and processing power. Both the embedded application and email server must be co-existent and must be running concurrently with proper interface between them. The mail server must be sleekly and should be small enough that it will operate within the limited resources after accommodating the embedded application code. It is necessary that many of the overheads existing in the protocol used for communication must be removed and elegant server must be designed and implemented. The email server must be interfaced with the ES application so that actions are taken based on the emails received from remote location.

III. Limitations:

In all the methods suggested in the literature, no attempt is made to reduce the size of the code. In fact, more of the code is added to provide more of the features such as authentication, trust, spam etc. Very little attempt has been made to port the implementation of any of the email server on the embedded system side.

IV. Literature Survey

[1] The SMTP protocol is mainly used for transmitting mail most reliably and efficiently and this protocol is not depend on specific transmission subsystem and desires just good information stream channel.SMTP design can be used for both one way communication and two way communications. Once the transmission channel established SMTP sender will send a mail command if the SMTP receiver accepts the mail it responds with ok reply if not it responds as a rejecting.

[2]. SMTP has given a steady, competent reason for the carry function of message exchange handlers. Defining a frame work for extending the SMTP service whereby a server SMPT informs the client SMTP with regards to the service increase it supports. [3] On specific sorts of littler hubs in the web is frequently unreasonable to keep up a message transport framework (MTS).

[4]. A MIME version header field which uses a version number to declare a message to be conformant with MIME

A content type header field generalized from RFC 1049, which can be used to specify the media type and subtype of data in the body of the message.

A content transfer encoding header field which can be used to specify both the encoding transformation and the domain of the result.

Two other header handle that can be utilized to additionally represent the information in the body, the content- ID and content depiction header fields

[5] Mainly presents the most widely used protocols for transmitting and receiving mails.

A detailed model is suggested for the estimation of upper and lower boundaries of volume generated by

May – June 2020
ISSN: 0193-4120 Page No. 2362 - 2365

SMTP which it can easily continue for POP3 protocol. Regarding end to end delay X400 is uniformly or more active than SMTP. In considering the parameters like volume build and frame length sharing, POP3 is more efficient even its functionality is limited. Whenever we maintain the TCP connection until all the messages are delivered then we can say that SMTP is improved in performance.

[6] Host grouping strategy that it reduces the delivery time of the messages queue to multiple hosts by processing all messages belonging to the same host mail server as one bulk.

[7] Email frameworks are ordinarily utilized because of their effortlessness, adaptability, and low expenses for their execution and use. In any case, such frameworks endure issues brought about by fragilities in the conventions required in the correspondence. Among them, there are issues concerning the nonappearance of a powerful instrument for sender confirmation, poor classification

What's more, respectability instruments for message conveyance, and furthermore the lacking of a reliable notoriety system for clients and email servers. This paper proposed and assessed a trust show for email servers. It characterizes trust amasses whose individuals connect to trade "conclusions" about outside email servers. Every server in a trust assemble utilizes the other individuals' feelings to construct a worldwide trust, and uses this data to restrict the amount of email got from an outside server. Nearby trust data is engendered to the trust amass utilizing an interpersonal organization demonstrate, decentralizing the support and development of trust data.

[8] In this an communication of email using protocols those are SMTP and POP3 and IMAP4 so for transporting emails from servers to clients we are using SMTP protocol and for receiving we are using pop3 and imap4 protocols are using and also this paper presents the an active monitoring algorithm to improve working efficiency of protocols and also to detect the failures occurring between while sending and receiving mails using these three protocols.

The working of active monitoring algorithm is in two ways first is to monitor the protocol server capacity to accept the new TCP formation requests on particular protocol port based and second one is to monitor the server whether it is able to accept approaching commands from the client and respond back with the appropriate response.

V. Limitations of the Existing Findings for Implementing Email Embedded Server

In all the methods suggested in the literature, no attempt is made to reduce the size of the code. In fact, more of the code is added to provide more of the features such as authentication, trust, spam etc. Very little attempt has been made to port the implementation of any of the email server on the Embedded system side.

VI. The Need for Implementing Email Server within an Embedded System

Embedded systems are being used extensively as data collection agents and also as systems that can receive the data and act on it for controlling the environment that it is operating. The instructions / Commands as such are required to be transmitted from over long distances through HOST based systems generally the PC and using internet. Communication with the embedded system can be effected using the internet in many ways which include Email exchange, Data communication through a HTML text, a WEB service request, a File transfer etc. Among all email is light weight and therefore is possible and fittest method for communication over the internet.

Thus, there is a need to implement email server within the embedded systems.

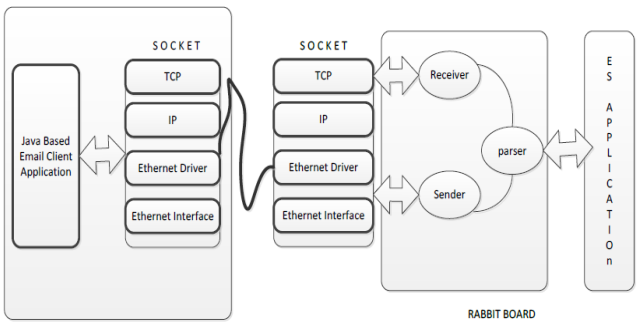
VII. Proposed Architecture for Implementing Email Server

One of the main features that have been avoided in building the email server is elimination of all the mechanisms that are needed for store and forward as

May – June 2020
ISSN: 0193-4120 Page No. 2362 - 2365

the emailing system that has been complied is point to point. Java based socket communication has been used on the client side and Micro C sockets are used on the Rabbit board side. The sending and receiving processes are implemented as sockets that encapsulate the TCP/IP address of the client or the server. The email server is comprised of three simple segments that include Receiver, sender and a Parser. Parser encodes an incoming email that is composed of commands and using the command one of the ES based application is invoked. Similarly while sending, the parser is triggered which forms an email and encapsulates the same into a socket before sending the same to the client. The entire ES application and email system is implemented under µCOS operating system. The proposed architecture for implementing email server is shown in figure.2.

By practically when we are implementing email server at any embedded system. The devices which we want to collect information are connected to one embedded board having TCP/IP configuration. This connection may be wired or wireless. This board may act a server. Server which is having all the data received from different devices as a records. Whenever client may request the data from the server it will send a request, with the response to that request server will send the corresponding information in the format of E-mails.



VIII. Conclusion

Communicating with an embedded system from a remote PC connected on to Internet can be achieved through implementation of email systems on the imbedded system. All the overhead related to the store and forwarding of the email messages is eliminated in the proposed architecture. A parser mechanism is added for encoding and decoding the email messages. The parser is interfaced with the ES application through implementation of a signalling process supported under RTOS.

References

1. Jonathan B. Postel, Simple Mail Transfer Protocol, The internet society, 1982
2. J. Klensin, SMTP Service Extensions, Network Working Group, 1995, 1 to 11
3. J. Myers,M.Rose, Post Office Protocol - Version 3, network working group, 1996
4. N.Freed,N.Borenstein, Multipurpose Internet Mail Extensions, Network Working Group, 1996, 1 to 31
5. P. Tzerefos, C. Smythe, I. Stergiou and S. Cvetlkovic, A Comparative Study of Simple Mail Transfer Protocol (SMTP),Post Office Protocol (POP) and X.400 Electronic Mail Protocols, IEEE, 1997, 545 to 554
6. M. H. Haggag, ENHANCED DELIVERY THROUGH A SMART SMTP CLIENT, Computer Science Department,Faculty of Computers and Information, Helwan University, 2004, 112 to 124
7. Leonardo B. de Oliveira and Carlos A. Maziero, A Trust Model for a Group of E-mail Servers, CLEI ELECTRONIC JOURNAL,, 2008, 1 to 11
8. R. Sureswaran, Hussein Al Bazar, O. Abouabdalla, Ahmed M.Manasrah, Active E-mail System Protocols Monitoring Algorithm, IEEE, 2009,1 to 6