An Optimized Adhoc Network for Real Time Applications

Binu C T

DOI: 10.29322/IJSRP.12.12.2022.p13222
http://dx.doi.org/10.29322/IJSRP.12.12.2022.p13222

Paper Received Date: 25th October 2022
Paper Acceptance Date: 1st December 2022
Paper Publication Date: 13th December 2022

Abstract- Adhoc Network for real time application require highly secure infrastructure. The real time application require data in its derivative forms. Security mechanism increases adhoc network in its actual form. The connection of smart devices such as mobile phones, iPad etc. to the internet result in the improvement of computation everywhere. The computational capability of devices are very important while using computation to solve problems. The existing systems have problems with scheduling of tasks while doing high speed adhoc network with security. The proposed system have unique registration number of which 6 digit given by the user and 4 digit by the system and scheduling is based on this unique number. There is a Registration layer in each devices. The last 4 digit in the registration number defines the capacity of the device to do computation. A new Sorted Encoding Encryption and Decryption is to provide information security to the devices

Index Terms- data structure, information security, adhoc network, real time system

I. Introduction

Stock Quote Update require more amount of RAM to run the application. The fetch and decode the data is happening in a fraction of seconds. Another important stage is execute the application. There is the important of RAM. The changes in data is happening in a fraction of second. High computation operations are happening in RAM.

Flight information system is a real time system which updates information about flights. This is also a high speed application and faster RAM is required to execute the application. Real time system require less response time compared to general systems. So faster RAM is required to run the application.

II. DIFFERENT TYPES OF NUMBER SYSTEMS

Binary is a type of representation of data in which the computer can understand. The data stored in the RAM in the form of binary.

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111
16	10000
17	10001
18	10010
19	10011
20	10100

Fig 1: Binary values with Decimal

ASCII is a type of representation of data. We can convert these ASCII value to binary. The data is stored in RAM in the form of binary.

I. TYPES OF RAM

1. *DIMM(Dual Inline Memory Module)* Compatible only with desktop computers.
2. *SO-DIMM(Small Outline Dual Inline Memory Module)* Compatible only with laptops and other portable devices.
3. *SRAM (Static RAM)* In an SRAM chip, each memory cell stores a binary digit

(i.e., 0 or 1) for as long as power is supplied. It is also larger in size.

4. *DRAM (Dynamic RAM)*
In a DRAM chip, the charge on individual memory cells must be refreshed periodically to retain data.

RAM can also vary depending on its age. DDR5 (Double Data Rate 5) is the most current. Generally, the processors and motherboards that are manufactured today are only compatible with the RAM of this generation. On the other hand, there are those of the generation DDR4 or DDR3. They are used by computers of a certain age. Although their presence is decreasing, they still circulate. This allows users of old computers or devices to expand their memory. With this, you can extend its lifespan without needing to make big payments.

III. OBJECTIVES

The objectives of the system to provide high speed adhoc network with security for real time application.

IV. DATA STORED IN RAM USING HASH TABLE

The main three operations in RAM are

- Search
- Insert
- Delete

Hash table is a data structure mainly used to store data in RAM. There is a key for the table to store data in RAM. The key is used to Search (key) data in RAM using probability of presence of data. Generate key to Insert () data into the RAM and the key may be a random number. Initially allocate a slot in RAM before going to insert data. The key is generated based on the custom of usage. More used data have a Rank and data is stored based on that Rank. More used data have highest rank. The Delete (key) operation is related to search operation .Initially find out the data and free the memory. In C++ new is the keyword to allocate memory and delete is the keyword to free the memory.

V. OPTIMIZED DATA TRANSFER ALGORITHM IN ADHOC NETWORK

The proposed algorithm uses an equation to store data in RAM. The mathematical equation is connected with the data structure to map data in RAM. The mathematical equation is $x^{c+1}+y^{c+1}\pm c=d$, where x and y are the data to represent actual data and C is a constant value ranges from 0 to n and finally d is the actual data. If the binary data are of 2 bits then c value is 0. So the equation must be x+y+0=d. Find the value of x and y and stored in RAM. For example, if the data is 4(0100) c value must be 1.

C	Data Ranges	Equation
0	0 to 4	$x+y\pm c=d$
1	0 to 15	$x^2+y^2\pm c=d$
2	0 to 32	$x^3+y^3\pm c=d$

Find in Network(Key k1,k2) where k1 and k2 are keys
for (i=0 to k1)
A[k1,i]
for (j=0 to k2)
A[k2,j]
Substitute the value of A[k1,i] and A[k2,j] in the corresponding equation to represent data and get the actual data.

Insert in Network(data d)
Allocate two slot in RAM for x and y and two for the two keys
Compute the value of x and y using the equation $x^{c+1}+y^{c+1}\pm c=d$
Call Random () for data d
The function random returns two key values to map x and y

Delete from Network(key k1, K2)
Search for x and y using Find (key k1, K2) function
If x and y are not null
Free x and y

KETWORDS: Information security, Scheduling, Computation

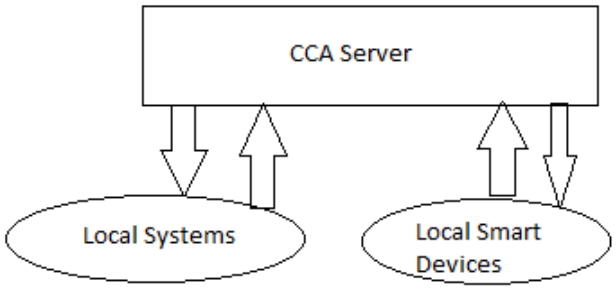

Fig 1: Optimized Adhoc Network in fight control area

VI. OPTIMIZED ADHOC NETWORK ARCHITECTURE

The proposed architecture provide high security and high speed computation. The figure shows the block diagram of Capacity Computing Architecture (CCA).

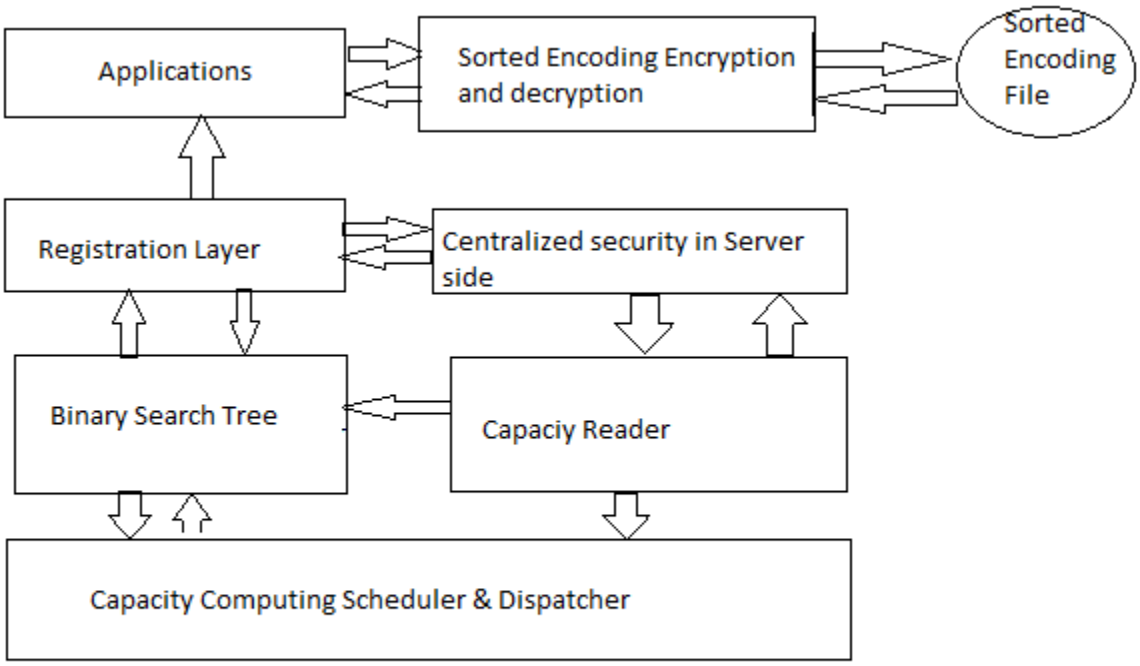

Fig 2: Architecture of Optimized Adhoc Network

The architecture diagram of the CCA devices connected to the system consist of Registration layer at the devices, Sorted encoding encryption and decryption method, Sorted Encoding file and the other applications. The server consist of centralized security in server side, capacity reader and capacity computing scheduler and dispatcher.

VII. REGISTRATION LAYER

Each device connected to the system have a registration number and it's a ten digit number. The six digit of registration number is given by the user .The last four digit of the registration number shows the capacity of computation in each devices which is set by the application .The registration layer is present in each devices manages the computation.The registration layer send the registration number to the server when the device is idle.The registraton layer connects with centralized security system in server to provide security.The registration layer provides information security to the other application in the device. Computation is done through the Registration layer.

VIII. .SORTED ENCODING ENCRYPTION AND DECRYPTION

The CCA provides security to the system connected with the system by using Sorted Encoding Encryption and Decryption method. Each file in the device is protected by the above method. The files in the device is encoded using Sorted Encoding file. The Sorted encoding file is updated each time when the user add new file in the device. The all files are decoded when the user click on the file. The Centralized Security in Server side blocks the intrusion and penetration in the file system.

IX. .SORTED ENCODING FILE

There is a sorted Encoding file present in the device which protects the files in the device. There is a 32 bit key is used to encrypt the sorted encoding file. The registration number is used as the 32 bit key for encryption. A random number is generated by the Encoding Encryption and Decryption and added to the Sorted Encoding file and make it as sorted. The random number is added after 128 bits 0f data. Sorted Encoding file is encoded with all the files in the device. The sorted Encoding file is sorted all the time and decoded when authorized user open the file. The Sorted Encoding file consists of unique password for each file. The file is automatically verifies the password when authorized user open the file.

X. .CENTRALIZED SECURITY IN SERVER SIDE

Centralized Security in Server side connects with Registration layer of each device. Centralized Security in server side have all the registration number of each device. It updates when new user is added with the system. Centralized Security in server side is connected with Capacity Reader. The Centralized Security in Server side blocks the intrusion and penetration in the file system

XI. CAPACITY READER

The capacity reader reads the registration number of devices which are free in current time. The last four digit of registration number is assigned to capacity. Capacity Reader is connected with the scheduler and dispatcher. It also updates each time when new message is reached in the Capacity Reader with a binary search tree in server.

XII. BINARY SEARCH TREE

Most of the system uses queue or priority queue for scheduling. The proposed system uses Binary search tree. The capacity Reader reads the capacity value of each devices and update in binary search tree. The scheduler reads the highest value each time when new message reaches the Capacity Reader. There is another type of tree called equation tree which can use if more security is needed. Each node have the capacity value. Equation tree uses linear equation or binomial equation to map the data into a tree.

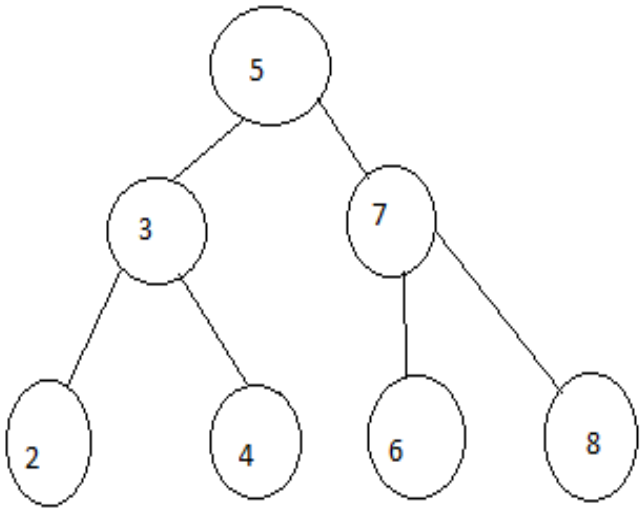

Fig 3: Binary Search Tree with capacity value

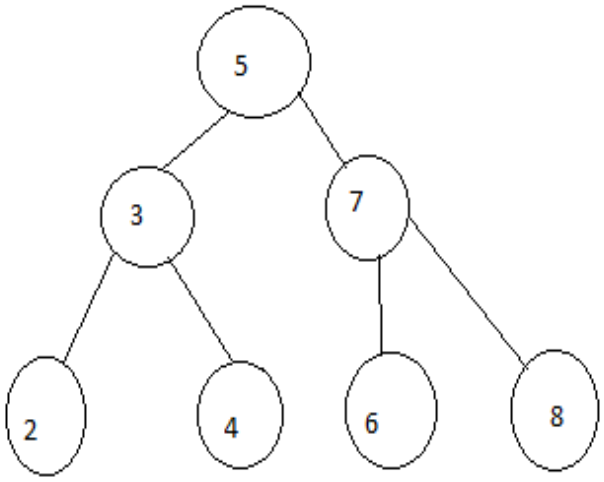

Fig 4: Equation Tree with adhoc network

XIII. SCHDULER AND DISPATCH

The proposed algorithm reads the capacity value from the binary search tree or from the equation tree and schedule the task for the computation to the device with highest capacity. The Earliest Finish Time is computed before doing the monitoring for each task. The task is represented in DAG (Directed Acyclic Graph).

1. for each Task Ti
2. Get Capacity(Ck) from Binary Search tree
3. Compute EFT
4. Initialize Time_free=EFT (Tj)
5. For time_free!=0
6. Select Dj with Ck high
7. Assign task to Dj
8. End if
9. Decrement Time_free
10. Compute_EFT(Ti,Dj)
11. End
12. Update free_time
13. End

CCA Algorithm

XIV. EXPECTED OUTCOME

The proposed Optimized Adhoc network touches the information security services and high performance is given to the System

AUTHORS

First Author – Binu C T, binuct143@gmail.com