Secure Data Transfer and 802.11 Channel Usage in Networking

Divya M V [1], G S Girish [2]

[1,2] *Department of Information Science Engineering,*
BNM Institute of Technology, Bangalore, Karnataka, India

Available online at: www.ijcseonline.org

Abstract— Data exchange rate is an estimation of the measure of information sent between two focuses on a system in a given time period. It is a profoundly critical idea in cutting edge business organizing, with high information exchange rates permitting systems to be utilized for complex undertakings, for example, internet spilling. Understanding information exchange rate could offer you some assistance with improving the execution of your business' own particular system. Today remote sensors system is broadly utilized as a part of all the association for their day-by-day correspondence. A remote sensors system is accumulation of hubs in a system. Each individual performs all their day by day exchange and correspondence with the assistance of web. In this paper we discuss mainly about Wi-Fi and its general utilization all over as it required no wire association it generally its work with the assistance of unguided media. The individual dependably imparts their data and information to the assistance of remote transmission some time they share private data, which required more security. As in the system, some unapproved individual expected to get to the individual and secret data of data.

Keywords—Remote Sensor, Wi-Fi, Remote Transmission.

I. INTRODUCTION

Driven by the expansion of Wi-Fi hotspots in broad daylight places, area based administrations (LBSs) have encountered surging improvement lately. LBSs exploit clients' area data to give customized or relevant administrations. A regular LBS framework comprises of a LBS supplier who offers administrations taking into account clients' physical areas through trusted Access Points (APs), and portable clients who demand particular administration alongside their own area and personality (ID) data [1].

  Remote systems normally have a lot of adaptability, which can interpret into quick organization. Remote systems utilize various base stations to interface clients to a current system. The base side of a remote system, nevertheless, is subjectively the same whether you are interfacing one client or a million clients. To offer administration in a given territory, you require base stations and reception apparatuses set up. Once that foundation is assembled, notwithstanding, adding a client to a remote system is for the most part a matter of approval. With the framework constructed, it must be designed to perceive and offer administrations to the new clients however, approval does not require more base. Adding a client to a remote system is a matter of designing the base, however it doesn't include running links, punching down terminals, and fixing in another jack.

  Remote gadgets are obliged to work in a specific recurrence band. Every band has a related data transmission, which is essentially the measure of recurrence space in the band. Transfer speed has gained an undertone of being a measure of the information limit of a connection. A lot of science, data hypothesis, and sign handling can be utilized to demonstrate that higher-transfer speed cuts can be utilized to transmit more data. As a case, a simple versatile telephony station requires a 20-kHz data transfer capacity. Television signs are tremendously more perplexing and have a correspondingly bigger transfer speed of 6 MHz The utilization of a radio range is thoroughly controlled by administrative powers through authorizing forms.

802.11 is an individual from the IEEE 802 family, which is a progression of determinations

 IEEE 802.11ac is a developing standard of Wireless Neighborhood (WLAN) that has accomplished Very High Throughput (VHT). VHT is accomplished with the assistance of efficient Modulation and Coding Schemes (MCS, for example, 256-Quadrature Amplitude Modulation (QAM), unequivocal transmit beamforming, upgraded Multiple Input Multiple Output (MIMO) innovation, and expansive data transmission. The majority of the main sellers and producers have as of now included 802.11ac in their Wi-Fi chipsets [2]. The 802.11ac works in 5 GHz band what's more, can backing a high information rate up to 6.933 GHz. Up to now the exploration has tended to concentrate on improvement.

        The Fig 1 represents the System Architecture where a encoded file can be decoded using Base64 and extract a csv file. Where the csv file would be loaded to the database using the csv loader. The file uploaded to DB can be fetched directly and displayed on the dashboard after filtering the special character.
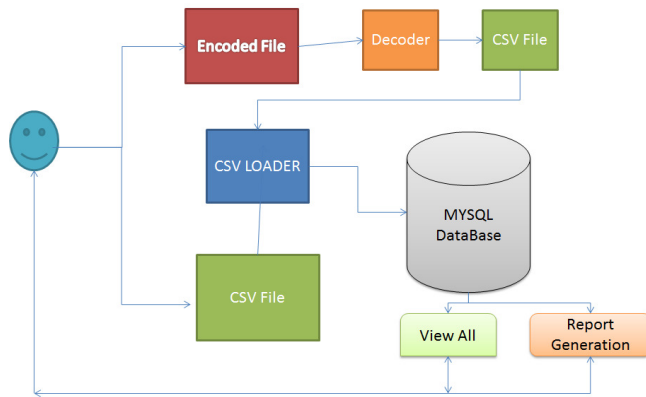
Fig 1 System Architecture

.

II. RELATED WORK

In execution administration, assessed information which was put away at first in the xml arrange. Putting away information in the xml arrange cause a huge increment in information size and preparing time. XML rehashes each component and characteristic name for each component and trait occurrence, The greatest execution hazard for XML comes not from the way that it is content based, that it is parsed, or that it can utilize Unicode yet from the way that XML archives can incorporate outside records. On the off chance that the information is in encoded shape then the time utilization to translate the information would be high .The decoder created utilizing either dialects like java, Perl and so forth., would be additional tedious. Execution administration (PM) incorporates exercises which guarantee that objectives are reliably being met in a successful and effective manner[3].Performance administration can concentrate on the execution of an association, an office, worker, or even the procedures to assemble an item or administration, and additionally numerous different regions.

III. DECODING

Base64 is a gathering of comparable twofold to-content encoding plots that speak to paired information in an ASCII translating so as to string group it into a radix-64 representation. The term Base64 begins from a particular MIME content exchange encoding[4].

Step 1: Get rid of those pesky equal signs:
To disentangle b64 code, the principal thing that you need to do is dispose of the equivalent sign(s) that might possibly be toward the end of the base64 code. On the off chance that you have a project that translates b64, you would need to leave those irritating equivalent signs or else the system would go insane and release a truly peculiar answer.

Code:

```
bW9uZXk=
b          W          9          u
Z          X          k
```

Step 2: Get the b64 decimal value
All you have to do now is get the b64 decimal value from the b64 chart.

Code:

```
bW9uZXk=
b          W          9          u
Z          X          k
27         22         61         46
25         23         36
```

Step 3: Convert to 6-bit binary
First take each b64 decimal number and change over it to 6-bit twofold. You should recollect, be that as it may, to change over it into 6-bit and NOT 8-bit paired. If you somehow managed to change over it into 8-bit twofold, you would get some bizarre answer, and it is conceivable that you wouldn't have the capacity to completely change over the b64 string again into ordinary content. At the point when changing over to 6-bit, you would do practically the same thing as you would for 8-bit, however leave off the main two digits. In the event that the encoding and initial 2 disentangling steps have been done accurately, the initial two digits of 8-bit twofold shouldn't contain a 1; they ought to both be zeros, so it wouldn't make any difference when you abandon them off in any case. So simply don't change over it to 8-bit, and after that you won't need to stress over anything that I said in the keep going (practically keep running on) sentence

Code:

```
b
27
011011
```

-Once all of the conversions have been made, my text editor looks like this:

Code:

```
bW9uZXk=
b          W          9          u          Z
X          k
27         22         61         46         25
23         36
011011     010110     111101     101110
011001     010111     100100
```

Step 4: Convert 6-bit to bit stream, then convert to 8-bit

The first thing that we're going to do is convert the 6-bit binary into one bit stream. After doing so, my text editor looks like this

Code:

```
b         W         9         u         Z
X         k
27        22        61        46        25
23        36
011011    010110    111101    101110
011001    010111    100100
0110110101101111011011100110010101111001
          00
```

Part the bit stream into 8-bit twofold. Every one of that implies is put a space or two (I lean toward two) between each eighth digit. Recall that, you ought to realize that there may be additional digits left over. They are the ones that must be put on when the string was encoded. In the event that none were left over, that implies none were included. On the off chance that there are any left over, they ought to just be zeros and they ought to fall off the end. These can be erased (in any case, if there are any ones left over, you accomplished something incorrectly when encoding or deciphering. My content tool now resembles this

Code:

```
b         W         9         u         Z
X         k
27        22        61        46        25
23        36
011011    010110    111101    101110
011001    010111    100100
0110110101101111011011100110010101111001
00
01101101  01101111  01101110  01100101
01111001
```

Step 5: Convert to decimal number

Now you're going to get the ASCII decimal number from the 8-bit binary.

Code:

```
01101101
64
32
 8
 4
+1
--
109
```

Once I find the ASCII decimal number for each 8-bit binary number, my text editor looks like this:

Code:

```
b         W         9         u         Z
X         k
27        22        61        46        25
23        36
011011    010110    111101    101110
011001    010111    100100
0110110101101111011011100110010101111001
          00
01101101  01101111  01101110  01100101
01111001
109       111       110       101
121
```

Step 6: Look it up

The ASCII table has four expansive sections which each contain five scaled down segment things. You will be utilizing the first and last smaller than expected sections in the greater part of the substantial segments. What you do is first take the number that you got after you changed over the 8-bit parallel. At that point find it under the small scale segment called "Dec" (that is the primary smaller than expected section in each huge segment). Take after the line until you get to the last small segment that is named "Chr". "Chr" remains for character, and "Dec" remains for decimal. So see what you're doing now? You're changing the DECimal into a ChaRacter. Goodness! After this is ruined the five decimal numbers that we have, my content tool resembles this:

Code:

```
b         W         9         u         Z
X         k
27        22        61        46        25
23        36
011011    010110    111101    101110
011001    010111    100100
0110110101101111011011100110010101111001
          00
01101101  01101111  01101110  01100101
01111001
109       111       110       101
121
m         o         n         e
y
```

-Now, to make it look a bit nicer:

Code:

```
b        W        9        u        Z
X        k
27       22       61       46       25
23       36
011011   010110   111101   101110
011001   010111   100100
011011010110111101101110011001010111001
              00
01101101  01101111  01101110  01100101
01111001
109      111      110      101
121
m        o        n        e
y
money
```

IV. CSV LOADER

The information which is brought from the decoder would be further investigated and evacuate the extraordinary character would be uprooted physically and stacked to DB.

Before importing the document, you have to set up the accompanying:

- A database table to which the information from the document will be foreign.
- A CSV document with information that matches with the quantity of segments of the table and the kind of information in every segment.
- The account, which interfaces with the MySQL database server, has FILE and INSERT benefits

V. EXPERIMENTAL RESULTS

The Fig 2.,represents the data which can be explained by taking an example of 2g and 3g users where the left part would indicate the number of 3G user and right side indicate the number of 2G user. The results can be displayed in the form of graph .
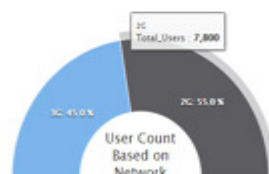

Fig 2 The percentage of users

VI. CONCLUSION

In this paper we are presented to another disentangling instrument where the information can be decoder inside of a small amount of second. The information which is encoded utilizing base64 decoder must be decoded by any individual. Along these lines structure document arrangement can be use for encoding an information with high Because of its permit cost its application is constrained to saving money, military and so on , which are very secure information. In coming up years we may even discover its application in considerably more zone to take the full fledge utilization of decoder

REFERENCES

[1] Cheng Chen, Rapeepat Ratasuk, Amitava Ghosh ," -" Downlink performance analysis of lte   and WIFI coexistence in unlicensed bands with a simple listen-before-talk scheme" vehicular technology conference (vtc spring), 2015 ieee 81st,2015

[2] Attiuttama and Kanojia Sindhuben babulal, "Analyzing Superiority of Tail Drop over Random Early Discard by evaluating throughput using varying queue length in 802.11", International Journal of Computer Sciences and Engineering, Volume-03, Issue-11, Page No (23-27), Nov -2015, E-ISSN: 2347-2693

[3] Zhenxing Liu1 , Lu Liu1,2, Richard Hill1 , Yongzhao Zhan2 "-"Base62x: An Alternative Approach to Base64 for non-Alphanumeric Characters", 2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)

[4] Gurpreet Singh, Supriya,-" Modified Vigenere Encryption Algorithm and its Hybrid Implementation with Base64 and AES", 2013 Second International Conference on Advanced Computing, Networking and Security

[5] Junhua Wang, Zhuoming Xu (), Jin Zhang"-" Implementation Strategies for CSV Fragment Retrieval over HTTP", 2015 12th Web Information System and Application Conference