# SIMPLE FILE SHARING USING SECURE AES ALGORITHM WITH END TO END CONNECTION

Harish Srinivasan[1], Vignesh .M[2], Santhi .G.B[3]

[1,2] Final Year CSE [3] Assistant Professor (CSE)

[1,2,3] New Prince Shri Bhavani College of Engg.

& Technology, Gowriwakkam, Chennai-73.

**ABSTRACT:**

File Sharing has turned out to be imperative to utilize and to work with data. End to End strategy for downloading permits clients to rapidly download a document or a file from any of the co-users yet here and there the information in associate gets disrupted and makes the client to download the adulterated/fake record and furthermore it's troublesome for the disrupted co-users to recoup their document or file back. So in this paper we are utilizing Secure AES Algorithm along with torrent technique for providing secure transmission of the file over a LAN or WAN.

**Keywords:** End to end connection, Torrent, AES.

## I. INTRODUCTION:

A large portion of the famous programming and entertainment downloads are in old customer/server downloading strategy. The issue here is while downloading an expansive record and when the interest for that document or file is high this will bring about moderate download speed in general and crash of server for the most part. So as to defeat this we will utilize end to end strategy for downloading (i.e. record exchange). Despite the fact that numerous end to end document or file sharing frameworks have been proposed and executed, just not very many have stood the trial of concentrated day by day use by an extensive client group. The Bit Torrent document sharing framework is one of these frameworks. Estimations have shown that Bit Torrent has developed into a standout amongst the most mainstream systems. It's the immense thought however why it is not actualized by any huge programming and diversion organizations?

## II. DRAWBACKS WITH THE EXISTING SCHEME:

Torrent has world wide users and hence it is used as a means for downloading and transporting files. But it has many security concerns such as data theft, data corruption and intrusion by third-party people, hence uploading a torrent file to the database can be considered unsafe.

## III.AES ALGORITHM:

AES was designed as a successor to the 3DES algorithm which was found to have certain limitations. When compared to 3DES, AES is more secure (it is less susceptible to cryptanalysis than 3DES).AES supports larger key sizes than 3DES's 112 or 168 bits.AES is faster in both hardware and software. And this algorithm is now used in real time applications like net banking, databases.

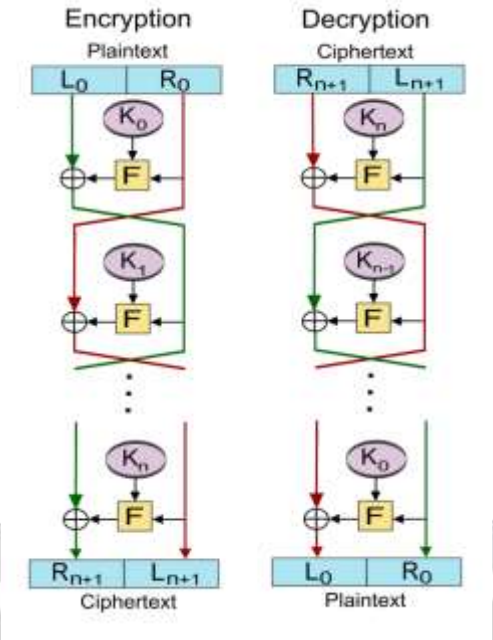


**FIG 1.1 AES ALGORITHM STRUCTURE**

**Encryption Process**

Here, we limit to portrayal of a normal round of AES encryption. Each round involve four sub-forms. The first round process is delineated underneath −

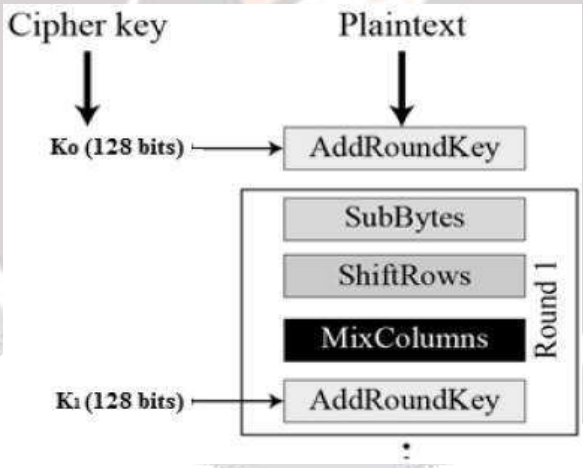


**FIG 1.2 AES ENCRYPTION STRUCTURE**

1.Byte Substitution (SubBytes)

The 16 input bytes are substituted by looking into a settled table (S-box) given in plan. The outcome is in a grid of four lines and four segments.

2. Shiftrows

Each of the four columns of the framework is moved to one side. Any passages that 'tumble off' are re-embedded on the correct side of column. Move is completed as takes after −

- First line is not moved.
- Second line is moved one (byte) position to one side.
- Third column is moved two positions to one side.
- Fourth line is moved three positions to one side.

The outcome is another lattice comprising of a similar 16 bytes yet moved as for each other.

3.

3, MixColumns

Every section of four bytes is presently changed utilizing an uncommon scientific capacity. This capacity takes as info the four bytes of one segment and yields four totally new bytes, which supplant the first section. The outcome is another new grid comprising of 16 new bytes. It ought to be noticed that this progression is not performed in the last round.

4. Addroundkey

The 16 bytes of the network are currently considered as 128 bits and are XORed to the 128 bits of the round key. In the event that this is the last round then the yield is the ciphertext. Something else, the subsequent 128 bits are deciphered as 16 bytes and we start another comparable round.

**Unscrambling Process**

The procedure of unscrambling of an AES ciphertext is like the encryption procedure in the turn around request. Each round comprises of the four procedures led in the turn around request −

- Add round key
- Mix segments
- Shift columns
- Byte substitution

Since sub-forms in each round are backward way, not at all like for a Feistel Cipher, the encryption and unscrambling calculations should be independently actualized, in spite of the fact that they are firmly related.

## IV. PROPOSED MODEL:

Overcoming the difficulties associated with the Bittorrent application, our model makes use of highly secure AES algorithm and it is accompanied by the creation of torrent files. The torrent generating software can be used to generate the torrent file which can be uploaded to the torrent database by a torrent client application after encryption and decryption.. The torrent file which was encrypted is sent to the clients present in the LAN or WAN network. The clients who have the decrypt code can access the torrent file. And finally the file can be transferred to the user without any loss of any data. The network traffic is completely low while downloading a torrent file. And thereby after establishing the end to end connectivity between the client and server, so that many clients can connect a server and access the content. Therefore the data is securely transmitted and could not be attacked by a intruder or hacker.

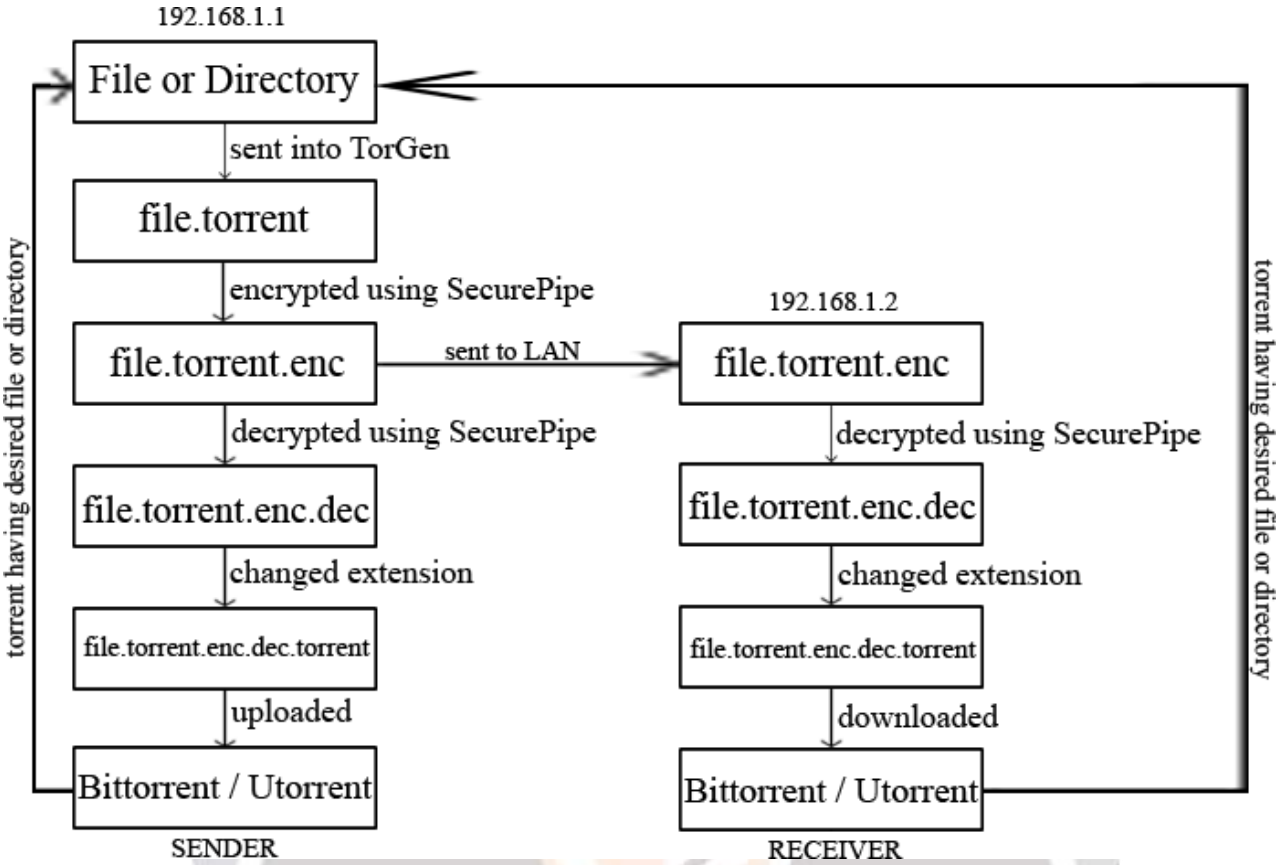


**FIG 1.3 ARCHITECTURE DIAGRAM**

**AT SENDER END**

The file or directory which is to be sent to receiver is fed into the TorGen tool. Then after entering desired contents to the respective file present in TorGen tool and after choosing several options the .torrent file is generated by the TorGen tool. Now the .torrent file which has the file or directory is open to all users, so in-order to secure it we hereby use the SecurePipe tool. The SecurePipe uses highly secure AES algorithm to encrypt and decrypt any kind of files or directories. And now we choose the .torrent file as a input to encrypt. After choosing the file we have to enter the password or the secret key. After entering the password or secret key, the .torrent file is completely encrypted by using AES algorithm. The same key will be used during the decryption of the file. Now SecretPipe generates the 'file.torrent.en' (for example) as of the architecture diagram. After encryption if you change the file extension and try adding the file as a torrent, then the Utorrent/Bittorrent will pop up several errors like you cant add a bencoded content. So the decryption of the file is much important before adding the file to a Bittorrent. The decrypt button is pressed for the file to be decrypted, the 'file.torrent.enc' file is chosen for the purpose of decryption. Then it becomes 'file.torrent.enc.dec' (for example) as of architecture diagram. And now the extension can be changed to 'file.torrent.enc.dec.torrent' and is ready to be uploaded to torrent database using a BitTorrent or Utorrent.

**AT RECEIVER END**

The encrypted file 'file.torrent.enc'(say) which is sent by the sender is received at the receiver end by the means of a LAN or a WAN. Now the receiver have to enter the same key which was entered by sender in-order to decrypt the file. And if he fails to enter the same key which is entered at the sender end, the file will not be decrypted. And so the file could not be downloaded successfully. Then after successful attempt of entering the key, the receiver downloads the file by changing the file extension and adding it into the Bittorrent or Utorrent.

## V. EXPERIMENTAL RESULTS:

Thus applying all those strategies for secure transmission of data over a network paved way some results. These results proved to be a successful one. Lets see what happens after torrent creation and downloading.

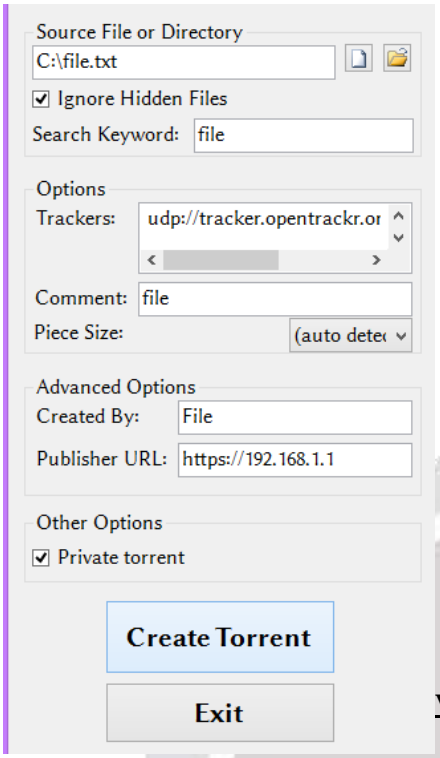


**FIG: 1.4 TORGEN TORRENTCREATION**

**FIG 1.5 AFTER DOWNLOADING USING TORRENT CLIENT**

## VI. CONCLUSION:

In this paper, we have seen creating a torrent file and sharing through a LAN and WAN using secure transmission. Here due to cryptographic algorithm implementation, the file is free from all kind of attacks. The attacker cannot predict how to encrypt the file or analyse what kind of algorithm used. The end to end connectivity provides endless connection between server and clients. Thereby reducing all kind of network traffic. This strategy can be applied from small scale to large scale networks. Only disadvantage is the key should be given to a trusted agent for the purpose of accessing the file. Switching to fiber optic cables will impact a high speed network transfers.

## VII. REFERENCES:

[1] M. Zeghid, M. Machhout, L. Khriji, A. Baganne, and R. Tourki, A Modified AES Based Algorithm for Image Encryption. International Journal of Computer Science and Engineering Volume 1 Number 1

[2] Morgan G. I. Langille, Jonathan A. Eisen, BioTorrents: A File Sharing Service for Scientific Data. Published: April 14, 2010.

[3] M.Pitchaiah, Philemon Daniel, Praveen ,Implementation of Advanced Encryption Standard Algorithm. International Journal of Scientific & Engineering Research Volume 3, Issue 3, March -2012 .

[4] Douglas Selent, ADVANCED ENCRYPTION STANDARD. InSight: RIVIER ACADEMIC JOURNAL, VOLUME 6, NUMBER 2, FALL 2010 .

[5] N. Bourbakis, A. Dollas, Scan-based compression-encryption hiding for video on demand. *IEEE Multimedia Mag*. 10, 79–87, 2003.

[6] Kaufman, C., Perlman, R., and Speciner, M. Network Security: Private Communication in a Public World. 2nd ed. Upper Saddle River, N.J.: Prentice Hall PTR, 2002

[7] Aakash Jasper ,Comparison of Local Area Network Technologies: Ethernet (IEEE 802.3), ATM and WLAN/WiFi (IEEE 802.11g). International Journal of Current Engineering and Technology E-ISSN 2277 – 4106, P-ISSN 2347 – 5161 ©2015 INPRESSCO® .