Aubrey gatewood 2/13/2024

Encrypt function:

I first open the key file to generate the round keys. Then I make a bitvector out of it and generate the key schedule. Then I made a bitvector from the plaintext file.

My while loop is true while there are more bits to read in the plaintext file. Then I initialize the state vector and do the initial xor with round-keys[0] element.

Then I start iterating over the 14 rounds. I ran subbytes, did the row shifts, then I had to convert back to a bitvector because I used numpy to do the array shifts. That was a mistake because it kept me from being able to change the steps for decrypt. After that, I do the matrix multiplication to get the MixColumns operation done. After that I xor with the round key, permute statearray to be prepared for the next round, and then I add the hex string of the bitvector to an output variable called tempstring. At the end of the 14 rounds I output tempstring to the ciphertext file and this is my output:   
  


Decrypted I could not get to work because I did a bad job managing my data types doing encrypt, so when I went to do decrypt, moving the steps around caused an unholy number of type errors