HW2
Vivek Khanolkar
vkhanolk
2/4/2021

Problem1:

For this problem I implemented DES based on the Feistel function. To achieve this I followed along with the steps in the Lecture 3 notes, as well as utilized code from the Lecture 3 section as well. The steps involved :

- Getting the encryption key
- Generating the round keys
- Reading in the file as a BitVector
- Running the feistel function 16 times
    - Permute the right side
    - XOR it with the round key
    - Perform S-box substitution
    - Perform P-box permutation (matches which input bit is corresponding output bit)
    - Add two halves together again
- Perform one extra swap to realign again

The same code was used to decrypt except I just had to call the round_keys in reverse.

For the key ‘zoomzoom’ the encrypted message was:

`J`A``Ul ```wy\c`YR```:@`
``[M``ʜvR`
`-``Y`+`<e```#``lN-`_``y``b```}txO`
``L``
`jU?`JyB
`````!T`````-er`
`h`j``Hb``|`w`n```r`````Da>``````|`pR``jB`q`!n)jT``}3`P#`,``"`
`cBz`.`y{```Eŕ`a`/`````j1;``0
0A``?'?`````rLd!``&``+``G`EQ`F`mq0`
-```Jo``O`7`ʕ`H`.`,```EC2t}v`濾aEE`>j``b`h`z`Oquū`
`o`y``6k/`:T```d`[fA`j,`y`
`夤=``7X`R`````e`P`Y``I`5B#,``@``Ł`
[```.`N`g`j\`A`g```IA{Wj`H```h`r```aR```G{`Ƀy````f`
`````

f'b n;C U H DV G L i" 1 R p , / 8m [} b
& Zw{B
q KB 4D ; F p@ C 6RA w hM) * Bpn u R A K]
eB з \8j,H ,un
k H*I B/ C`X& m , g Q : s X XwR uk
k[|6 YH e7 xW! x 2].w NY ul ! I
f Y T^ 3PmU- . ^ η
C h hI 0 FuXq iw7 % s jp + j T
Pq F F* eN Z/ # 5
1 /G >?1 |O{_ +U K kxys3u T C y h <Ɛ M a 5! f
PY Z0 f tO 'r f \M f F, CNŒF f 3
!+ AX Xx & з 4Z' 6 E 0 L T ^? 3 J 0 XU ~ o
a$ 6g}] ی Ū H s M Y }L5 y8(.710 R u

The decrypted message was:

Smartphone devices from the likes of Google, LG, OnePlus, Samsung and Xiaomi are in danger of compromise by cyber criminals after 400 vulnerable code sections were uncovered on Qualcomm's Snapdragon digital signal processor (DSP) chip, which runs on over 40% of the global Android estate. The vulnerabilities were uncovered by Check Point, which said that to exploit the vulnerabilities, a malicious actor would merely need to convince their target to install a simple, benign application with no permissions at all.The vulnerabilities leave affected smartphones at risk of being taken over and used to spy on and track their users, having malware and other malicious code installed and hidden, and even being bricked outright, said Yaniv Balmas, Check Point's head of cyber research. Although they have been responsibly disclosed to Qualcomm, which has acknowledged them, informed the relevant suppliers and issued a number of alerts - CVE-2020-11201, CVE-2020-11202, CVE-2020-11206, CVE-2020-11207, CVE-2020-11208 and CVE-2020-11209 - Balmas warned that the sheer scale of the problem could take months or even years to fix.

Problem2:

For problem 2 I utilized the exact same DES method as in problem 1. The only thing I had to do was read in the first three lines of the plaintext ppm file (as ppm files have a header that does not need to be encrypted/decrypted) and write it to the output file before performing the encryption.

Here is the encrypted image: