Cryptography, Fall 2019 Home Work 2 Due: 11-10-2019

Q: 1: We study a real-world case in this problem. A commercial file encryption program from the early 1990s used standard DES with 56 key bits. In those days, performing an exhaustive key search was considerably harder than nowadays and thus the key length was sufficient for some applications. Unfortunately, the implementation of the key generation was flawed, which we are going to analyze. Assume that we can test 10^6 keys per second on a conventional PC.

The key is generated from a password consisting of 8 characters. The key is a simple concatenation of the 8 ASCII characters, yielding $64 = 8 \cdot 8$ key bits. With the permutation PC−1 in the key schedule, the least significant bit (LSB) of each 8-bit character is ignored, yielding 56 key bits.

1. What is the size of the key space if all 8 characters are randomly chosen 8-bit ASCII characters? How long does an average key search take with a single PC?
2. How many key bits are used, if the 8 characters are randomly chosen 7-bit ASCII characters (i.e., the most significant bit is always zero)? How long does an average key search take with a single PC?
3. How large is the key space if, in addition to the restriction in Part 2, only letters are used as characters. Furthermore, unfortunately, all letters are converted to capital letters before generating the key in the software. How long does an average key search take with a single PC?

Q: 2: Take a plain text (At least 64 characters with 32-bit block size) and encrypt with 3-rounds DES. Show that robustness of cipher increase for frequency analysis by increasing number of rounds.

Q: 3: MATLAB/programming problem:
Design a S-Box for DES (4x4) and verify that for each input it holds Strict Avalanche Criterion (Change in single bit at input inverts half output bits).