Task3

Why finding K1 is enough to recover the full 64-bit DES key

In DES, the original 64-bit key includes **8 parity bits** at positions 1, 8, 16, 24, 32, 40, 48, and 64. These bits are not actually used in encryption. When we remove them, we get a **56-bit effective key**.

To generate the first round subkey **K1 (48 bits)**, DES performs these steps:

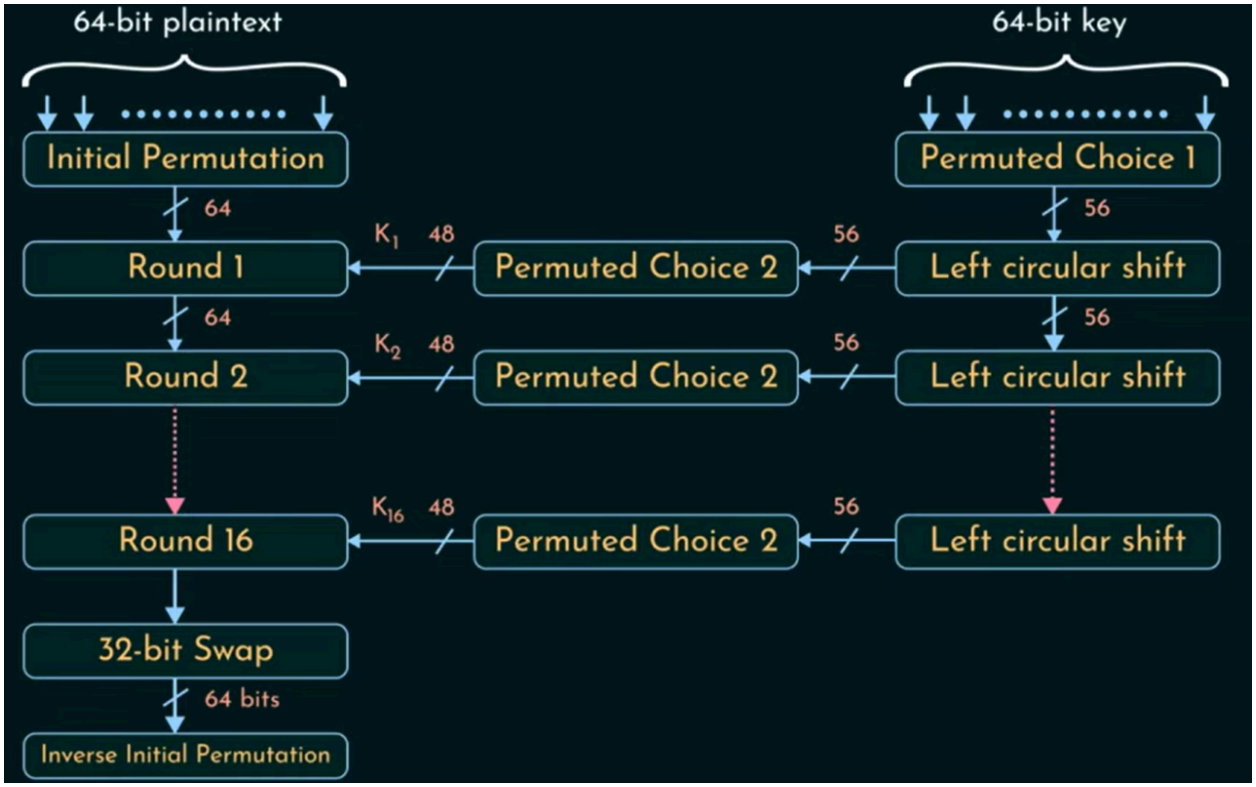
1. **Initial key permutation (PC-1):**
   This step rearranges the bits of the 64-bit key and discards the 8 parity bits, leaving a 56-bit result. This result is then split into two halves:
   - **C0:** the left 28 bits
   - **D0:** the right 28 bits
2. **Left circular shift:**
   For round 1, both halves **C0** and **D0** are rotated left by **1 bit** to produce **C1** and **D1**. This shifting changes the positions of bits slightly but does not lose any information, it's a reversible operation.
3. **Subkey permutation (PC-2):**
   The combined 56 bits (C1 + D1) are then permuted and reduced to **48 bits** using the PC-2 table. This step both reorders bits and removes 8 of them.

So, if we already know **K1 (48 bits)**, we can reverse these steps to find the original 64-bit key:

- **Step 1:** Reverse the PC-2 permutation. Since only 8 bits were dropped, we can brute-force those missing 8 bits. That means only $\mathbf{2^8 = 256}$ possibilities which is small and easily doable.
- **Step 2:** Reverse the 1-bit left shift to recover **C0** and **D0**. Because the rotation is circular, undoing it is straightforward.
- **Step 3:** Combine C0 and D0 to get the original 56-bit key, and then reinsert the 8 parity bits (which are unused) in positions 1, 8, 16, 24, 32, 40, 48, and 64.

That gives back the full 64-bit DES key.

In summary, knowing **K1** gives almost all the information about the key the remaining uncertainty is only 8 bits, which can be brute-forced easily.


64-bit plaintext
64-bit key
Initial Permutation
Permuted Choice 1
64
56
Round 1
K1
48
Permuted Choice 2
56
Left circular shift
64
56
Round 2
K2
48
Permuted Choice 2
56
Left circular shift
Round 16
K16
48
Permuted Choice 2
56
Left circular shift
32-bit Swap
64 bits
Inverse Initial Permutation