Discussion Question 6: Describe your communication protocol. Please refer to Section 2.2 for the components involved in a protocol.

We have two codes – sender.c and receiver.c.

The sender takes the input from the user and the receiver tries to decode the input. The sender and receiver process share the same L2 Cache. They run on an SMT as sender part of one thread and receiver part of the other thread.

The attack is Prime + Probe attack where the receiver first prime the L2 cache and the sender then access some memory address based on the user input and then the receiver again probe the L2 cache. If the time of block access is more than the configured threshold, it means the sender has accessed the corresponding cache line. Based on this info, the receiver can decode the input number.

Algorithm for Communication and Attack:

1. The user gives 8-bit integer as input. So, the values vary from 0 to 255.
2. The sender waits for the Enter Key from user and then the user enters the number.
3. Since the number is 8-bit, we target 8 unique sets for each bit.
4. We first allocate a huge page of 2MB so that the index bits for Virtual Address as well as the corresponding physical address is same which maps to the same cache set.
5. After allocating the huge page, we formed the eviction set for the sender.
6. Now, we allocate a buffer for sender and receiver equal the size of L2 cache. Now, we have the starting address of buffer.
7. In the eviction set we calculate the virtual address for which the index bits are the same as chosen for each bit.
8. The receiver also follows the same algorithm and find the eviction set which points to the same index as the sender eviction set.
9. So, here the receiver and sender work on 8 unique L2 sets, each having 8 cache lines. It means 64 cache lines. We have 64 addresses in the eviction set for sender and receiver.
10. Now, the receiver reads the address stored in eviction set, which is the prime process.
11. Then after the prime, the user gives the input to the sender.
12. The sender gets the input as string, convert it into binary.
13. Now, if the nth bit is 1, the sender accesses the particular cache lines for the corresponding eviction set.
14. If the nth bit is 0, the sender doesn’t access the cache lines.
15. The receiver probe cache sets corresponding to its eviction set and find the time to access the block.
16. If the access time for the particular bit is more than threshold, the corresponding bit is 1.
17. If the access time for the particular set is less than threshold, the corresponding bit is 0.
18. We use the clock function for both sender and receiver, where sender evicts the cache line for particular clock time and receiver also follows the same.
19. Now, receiver has the bit (1 or 0) for each set (8 sets in eviction set).
20. It converts the binary number to Integer and we have the decoded number.
21. The threshold for cache access time at the receiver side for my code is 70 cycles.