CIS5370-project2

prashant.ravi

January 2025

1 Demonstration

The Github repository contains a demo.mp4 to display the attack in video format and the followin image describes how one can attack the game score in Red Alert 2.

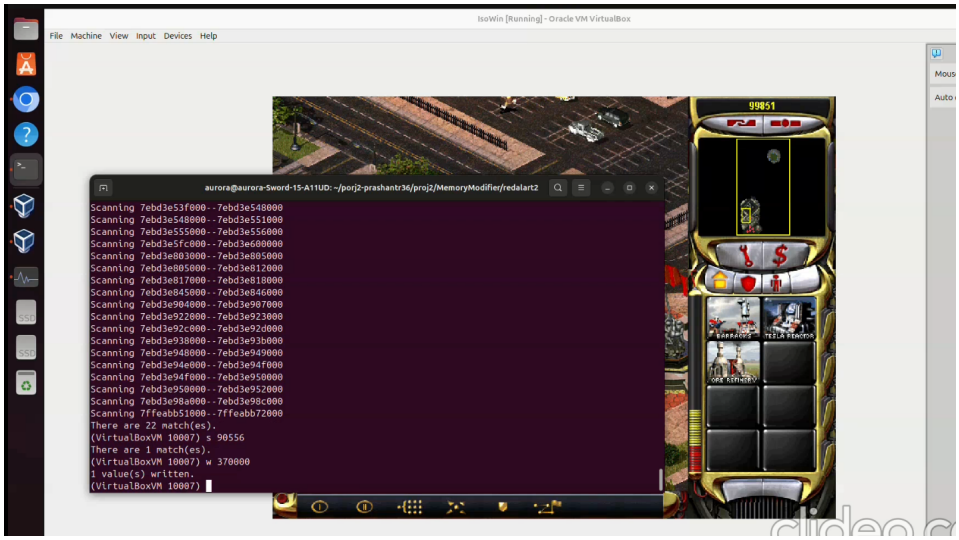

Figure 1: Gaining Shell access as user on STACK.FLE executable

2 Question 1

Modified the code to keep track of memory addresses that have been matched already and narrows down the search every time 's ¡value¿' is entered. Finally there will be only 1 matching memory address that can be updated from the command line which will reflect in the game as well. The previous iterative search was too slow and inaccurate because it doesn't remember the memory address matches across searches . The modification provided will search for the new values and also discard the previous matches that remained the same in the match.

As we can see in the demo video that the originally score is 500,000 and then we search for this value in the memory of the Red Alert game. After this we purchase Barrack and this results in the dollar amount decreasing and we search again for this new dollar amount in the memory modifier program. After the second result search we have finally found all locations that contains the new score and we modified the score to be the value 20 which can be observed in the top right hand corner of demo video provided.

Below are the modifications made in the code that discards the excessive memory locations in the remain hashmap and retrieve the desired memory addresses that will be modified by the 'w 20' instructions from the command line.

```
else {
          // Search in the watched values.

          for (auto it = remain.begin(); it != remain.end(); ) {
              uint32_t current_value = load(it->first);

              if (current_value != val) {
                  // If value is different from search value, remove it
                  it = remain.erase(it);
              } else if (current_value == it->second) {
                  // If value has remained the same since insertion, remove it
                  it = remain.erase(it);
              } else {
                  ++it;
              }
          }
      }
      printf("There are %ld match(es).\n", remain.size());
}
```