AFL-INFER – OBSERVATIONS

Table of Contents

Program 1: Buggy Linked List C program

```
//linked list to print the values
#include <stdio.h>
#include <stdlib.h>

struct node{
  int val;
  struct node *next;
  struct node *prev;
};
struct list {
  struct node* head;
  struct node* tail;
};

struct list list;

void add_node(int val) {            //bug 1 (Described below)
  struct node* new_node = (struct node*)malloc(sizeof(struct node));
  if(new_node != NULL){
    new_node-> val = val;           //bug 1 (Described below)
    list.tail->next = new_node;
    list.tail = new_node;
    new_node->prev = list.tail;
  }
}

void print_list() {                 //bug 2 (Described below)
  struct node* cur;
  for(cur = list.head; cur->next!=NULL; cur = cur->next) {  //bug 2 (Described below)
    printf("%d ",cur->val);
  }
  printf("\n");
  return;
}

void delete_node(int val) {
  struct node* cur = list.head;
  if(cur == NULL) {
    return;
  }
  while(cur!= NULL) {
    if(cur->val == val){
      if(cur == list.head) {
        list.head = cur->next;
      }
      if(cur == list.tail) {
```

```
      list.tail = cur->prev;
    }
    if(cur->next != NULL){
      cur->next->prev = cur->prev;
    }
    if(cur->prev != NULL){
      cur->prev->next = cur->next;
    }
    free(cur);
  }
  cur = cur->next;          //bug 4 (Described below)
 }
}

int main() {
 short int ch;
 int val;
 printf("1 - Insert a value\n");
 printf("2 - Delete a value\n");
 printf("3 - Print the list of values\n");
 printf("10 - Exit\n");

 while (1)
 {
    printf("Enter choice: ");
    scanf("%hd", &ch);    //bug 3 (Described below)
    switch (ch) {
      case 1:
        printf("Enter a value to insert: ");
        scanf("%d",&val);  //bug 3 (Described below)
        add_node(val);
        break;
      case 2:
        printf("Enter a value to delete: ");
        scanf("%d",&val);  //bug 3 (Described below)
        delete_node(val);
        break;
      case 3:
        print_list();
        break;
      case 10:
        return 0;
      default:
        printf("Wrong menu choice\n");
    }
 }
 return 0;
}
```

Bugs in the Linked list Program:

(Highlighted in the above code)

- add node function doesn't handle the case that list could be empty. Line 21 crashes the program if list.tail is NULL.

- print list function has a similar bug; the cur->next on Line 28 crashes if list.head is Null.

- All the scanf invocations in the program make the program hang if a non-numeric input is provided by the user.

- curr in the line highlighted could be null.

Screen print of Infer results:

1) When no additional flag is used to run Infer, Infer finds 0 issues with the code.

```
osboxes@osboxes:~$ infer run -- gcc -c linkedlist.c
Capturing in make/cc mode...
Found 1 source file to analyze in /home/osboxes/infer-out
Starting analysis...

legend:
  "F" analyzing a file
  "." analyzing a procedure

F....

  No issues found
osboxes@osboxes:~$
```

2) When Infer is run with debug flag(*Infer –debug run –gcc –c linkedlist.c)*, it finds two errors.

```
Terminal File Edit View Search Terminal Help
Capturing in make/cc mode...
Found 1 source file to analyze in /home/osboxes/infer-out
Starting analysis...

legend:
  "F" analyzing a file
  "." analyzing a procedure
  "C" analyzer crashed
  "T" timeout: procedure analysis took too much time
  "S" timeout: procedure analysis took too many symbolic execution steps
  "R" timeout: procedure analysis took too many recursive iterations

F....Smake: Entering directory '/home/osboxes/infer-out/multicore'

make: Leaving directory '/home/osboxes/infer-out/multicore'

Found 2 issues

linkedlist.c:59: error: USE_AFTER_FREE (biabduction/Rearrange.ml:1655:50-57:)
  [B5] pointer `cur` last assigned on line 39 was freed by call to `free()` at line 57, column 7 and is dereferenced or freed at line
 59, column 11.
  57.         free(cur);
  58.       }
  59. >     cur = cur->next;
  60.     }
  61.   }

linkedlist.c:87: error: Assert_failure (biabduction/Match.ml:577:6-6:)
  .
  85.           break;
  86.         case 3:
  87. >         print_list();
  88.           break;
  89.         case 10:


Summary of the reports

                                    USE_AFTER_FREE: 1
  Assert_failure (biabduction/Match.ml:577:6-6:): 1
osboxes@osboxes:~$
```

- Error 1: USE_AFTER_FREE- It points out that a pointer is being used after its memory is freed. (Bug 4)
- Error 2: It would be really helpful if someone can elaborate on error:**Assert_failure (biabduction/Match)** as the error message is not very verbose.

Screen Print of AFL results:

```
                    american fuzzy lop 2.52b (linkedlist)

┌─ process timing ──────────────────────────────────────┬─ overall results ─────┐
│        run time : 0 days, 0 hrs, 26 min, 19 sec       │  cycles done : 21     │
│   last new path : 0 days, 0 hrs, 2 min, 4 sec         │  total paths : 15     │
│ last uniq crash : 0 days, 0 hrs, 23 min, 39 sec       │ uniq crashes : 5      │
│  last uniq hang : 0 days, 0 hrs, 26 min, 15 sec       │   uniq hangs : 3      │
├─ cycle progress ────────────────────┬─ map coverage ─┴───────────────────────┤
│  now processing : 2 (13.33%)        │    map density : 0.01% / 0.02%         │
│ paths timed out : 0 (0.00%)         │ count coverage : 3.67 bits/tuple       │
├─ stage progress ────────────────────┼─ findings in depth ────────────────────┤
│  now trying : splice 5              │ favored paths : 3 (20.00%)             │
│ stage execs : 84/128 (65.62%)       │  new edges on : 4 (26.67%)             │
│ total execs : 79.7k                 │ total crashes : 3253 (5 unique)        │
│  exec speed : 49.55/sec (slow!)     │  total tmouts : 74.8k (3 unique)       │
├─ fuzzing strategy yields ───────────┴───────────────┬─ path geometry ────────┤
│   bit flips : 1/288, 1/282, 0/270                   │    levels : 5          │
│  byte flips : 0/36, 0/30, 0/18                      │   pending : 9          │
│ arithmetics : 0/2016, 0/517, 0/0                    │  pend fav : 0          │
│  known ints : 0/184, 0/840, 0/88                    │ own finds : 14         │
│  dictionary : 0/0, 0/0, 0/0                         │  imported : n/a        │
│       havoc : 9/53.2k, 8/21.2k                      │ stability : 100.00%    │
│        trim : 5.71%/7, 0.00%                        ├────────────────────────┘
└─────────────────────────────────────────────────────┘          [cpu000: 12%]
^C

+++ Testing aborted by user +++
[+] We're done here. Have a nice day!
```

1) AFL generated 5 crashes. These inputs in the crashes, when fed to the program , redirects us to bugs 1&2.
2) AFL hangs point out bug 3 listed above.