NOEMI DE MARTINO

S2/L5

Debugging

Agenda

- Traccia
- What does the code do?
- Debug
- Attacco OVERFLOW
- Debugged code
- RUNNING code

Traccia

Per agire come un Hacker bisogna capire come pensare fuori dagli schemi. L’esercizio di oggi ha lo scopo di allenare l’osservazione critica.
Dato il codice in allegato, si richiede allo studente di:

- Capire cosa fa il programma senza eseguirlo.

- Individuare dal codice sorgente le casistiche non standard che il programma non gestisce (esempio, comportamenti potenziali che non sono stati contemplati).

- Individuare eventuali errori di sintassi / logici.

- Proporre una soluzione per ognuno di essi.

What does the code do?

Bugged code

Lo script di oggi è stato scritto usando il linguaggio C.
Anche senza eseguirlo, risulta essere un programma per un assistente digitale che è in grado di fare operazioni numeriche di base (es. moltiplicazione e divisione fra numeri interi in input dall'utente).
Sono però evidenti alcuni problemi e vulerabilità presenti nello script, che potrebbero influenzare il suo funzionamento.

Debug

Errori nella funzione int main():

- **Riga 14:** Scanf(“%d”, &scelta) **Errore di tipo sintattico**. Abbiamo un input in char non un int.

- **Da riga 16 a 26: Errore di tipo logico**. Nello switch non viene considerato il caso in cui l’utente inserisce qualsiasi altra opzione oltre a quelle indicate, va inserita l’opzione default per gestire le altre possibilità di input. Inolte non vengono tenuti conto il possibile inserimento delle lettere in minuscolo “a”, “b”, “c” in input.

```
11  {
12  char scelta = {'\0'};
13  menu ();
14  scanf ("%d", &scelta);
15
16  switch (scelta)
17  {
18  case 'A':
19  moltiplica();
20  break;
21  case 'B':
22                  dividi();
23                  break;
24  case 'C':
25                  ins_string();
26                  break;
27  }
```

Debug

Errori all’interno della funzione moltiplica():

- **Riga 45 e 50: Errore di tipo logico**. L’utilizzo di short int in una moltiplicaizone fra i numeri non è la scelta più adatta poichè short int è un rage limitato (un minimo di -32,678 e un massimo di 32,767). Meglio usare int.

- **Riga 47 e 48: Errore sintattico**. Viene usato “%f” invece che “%hd”.

```
43  void moltiplica ()
44  {
45  short int  a,b = 0;
46  printf ("Inserisci i due numeri da moltiplicare:");
47  scanf ("%f", &a);
48  scanf ("%d", &b);
49
50  short int prodotto = a * b;
51
52  printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto);
53  }
```

Debug

Errori all’interno della funzione dividi():

- **Riga 58 e 64: Errore di tipo logico.** L’inserimento di int all’interno di una divisione non è consigliato perchè il risultato potrebbe contenere un numero non intero. Sarebbe opportuno usare float.

- **Riga 60 e 62: Errore logico.** L’utilizzo di “%d” quindi non è correttto perchè ci aspettiamo un valore di tipo “float”.

- **Riga 61: Errore logico**. Non viene presa in considerazione l’ipotesi che un utente scelta come divisore 0

```
56  void dividi ()
57  {
58          int  a,b = 0;
59          printf ("Inserisci il numeratore:");
60          scanf ("%d", &a);
61  printf ("Inserisci il denumeratore:");
62          scanf ("%d", &b);
63
64          int divisione = a % b;
65
66          printf ("La divisione tra %d e %d e': %d", a,b,divisione);
67  }
```

Debug

Errori all’interno della funzione dividi():

- **Riga 64: Errore di tipo logico.** Il risultato sara di tipo float e non int, inoltre l’espressione scritta restituirà il modulo dei due numeri e non la divisione.

- **Riga 66: Errore logico.** “%d” inseriti qui non sono ovviamente corretti e dovrebbero essere sostuiti con “%f”.

```
56  void dividi ()
57  {
58          int  a,b = 0;
59          printf ("Inserisci il numeratore:");
60          scanf ("%d", &a);
61  printf ("Inserisci il denumeratore:");
62          scanf ("%d", &b);
63
64          int divisione = a % b;
65
66          printf ("La divisione tra %d e %d e': %d", a,b,divisione);
67  }
```

Debug

Errori all’interno della funzione ins_string():

- **Riga 76: Errore di tipo logico.** Non è presente un controllo sulla lunghezza dell’input, ciò mette a rischio il programma esponendolo ad un possibile **attacco overflow**.

- **Riga 77: Errore di tipo sintattico.** La presenza della “&” non è necessaria dato che il puntatore è già indirizzato verso la prima posizione dell’array, questa parte di codice non restituirà a schermo la stringa inserita in output.

```
73   void ins_string ()
74   {
75   char stringa[10];
76            printf ("Inserisci la stringa:");
77            scanf ("%s", &stringa);
78   }
```

Attacco owerflow

Un attacco overflow, o più precisamente un attacco buffer overflow, è un tipo di vulnerabilità di sicurezza che si verifica quando un programma tenta di scrivere più dati in un buffer di quanto questo possa contenere.

- Esecuzione di codice arbitrario: Un attaccante può sfruttare un buffer overflow per sovrascrivere parti della memoria del programma con codice maligno e far sì che il programma lo esegua.
- Corruzione dei dati: I dati vicini al buffer possono essere sovrascritti, causando comportamenti imprevedibili o crash del programma.
- Escalation dei privilegi: In alcuni casi, l'attaccante può ottenere il controllo del sistema con privilegi elevati.

Debugged code

Correzione errori

Maggiore accessibilità

RUNNING code

```
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
D >> Uscire
Inserisci qui la tua scelta: a
Inserisci il primo numero: 5
Inserisci il secondo numero: 4
Il prodotto tra 5 e 4 è: 20
Vuoi fare qualcos'altro? (S/N): s
```

```
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
D >> Uscire
Inserisci qui la tua scelta: B
Inserisci il numeratore: 8
Inserisci il denominatore: 7
La divisione tra 8.000000 e 7.000000 è: 1.142857
Vuoi fare qualcos'altro? (S/N): s
```

RUNNING code

```
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
D >> Uscire
Inserisci qui la tua scelta: c
Inserisci una stringa di massimo 10 caratteri: 456568
La stringa inserita è: 456568
Vuoi fare qualcos'altro? (S/N): S
```

```
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
D >> Uscire
Inserisci qui la tua scelta: d
Uscita dal programma.
```