

Engenharia de Segurança

Aula 11 - 30/04/2018

Afonso Fontes
(pg35389)

Bruno Carvalho
(a67847)

Mariana Carvalho
(a67635)

7 de Maio de 2018
Universidade do Minho

Pergunta P1.1 - overflow.c

1. Qual a vulnerabilidade que existe na função `vulneravel()` e quais os efeitos da mesma?

A vulnerabilidade da função encontra-se nos tipos dos argumentos `x` e `y`. Se os parametros `x` e `y` passados aquando da chamada da função `vulneravel()` forem superiores ao valor maximo do tipo `size_t`, que corresponde a um inteiro sem sinal, estes vão ser convertidos para valores menores e por isso serão alocados para a matriz menos bytes do que o pretendido, o que levará a que se tente escrever em posições de memória não alocada para a função.

2. Complete o `main()` de modo a demonstrar essa vulnerabilidade.

```
#include <stdio.h>
#include <stdlib.h>

void vulneravel (char *matriz, size_t x, size_t y, char valor) {
    int i, j;
    matriz = (char *) malloc(x*y);
    for (i = 0; i < x; i++) {
        for (j = 0; j < y; j++) {
            matriz[i*y+j] = valor;
        }
    }
}

int main() {
    char *m;
    vulneravel(m, 2147483700, 2147483700, 0);
}
```

O `main()` foi completado com valores superiores ao valor máximo do tipo `int` na máquina.

3. Ao executar dá algum erro? Qual?

Ao executar o código ocorre *Segmentation Fault*.

Pergunta P1.2 - underflow.c

1. Qual a vulnerabilidade que existe na função `vulneravel()` e quais os efeitos da mesma?

A função aceita qualquer valor de tamanho (desde que menor que 2048) e cria um buffer e copia o número de bytes respectivo sem nunca verificar se o tamanho efectivamente corresponde ao tamanho do buffer a copiar. Isto permite que um atacante passar à função um valor de tamanho maior do que o buffer a copiar e assim copiar indevidamente não só o buffer mas também as posições de memória contiguas.

2. Complete o `main()` de modo a demonstrar essa vulnerabilidade.

```
int main() {  
    char origem[] = {'o', 'l', 'a'}; //buffer com 3 bytes  
    vulneravel(origem, 2047); // mas a função vulneravel() vai copiar 2047 bytes  
    return 0;  
}
```

3. Ao executar dá algum erro? Qual?

Não, não dá qualquer erro ao executar.

Pergunta P1.3 - erro_sinal.c

Qual a vulnerabilidade que existe na função `vulneravel()` e quais os efeitos na mesma?

Neste caso a vulnerabilidade que existe está na assinatura da função `vulneravel()`, que recebe como argumento um parâmetro *tamanho*, sendo este interpretado como sendo do tipo *size_t*, ou seja, na prática um *unsigned int*. Par

3. Complete o main() de modo a demonstrar essa vulnerabilidade

```
int main() {  
    char origem[] = {'o', 'l', 'a'}; //buffer com 3 bytes  
    vulneravel(origem, -10);  
    return 0;  
}
```

3. Ao executar dá algum erro? Qual?

Dá um Segmentation fault.