プログラミング工学・演習課題

工学部情報工学科 4619072 服部翼

2019 年 12 月 25 日

1　バッファオーバフローとは

まず、「バッファ」というのはコンピュータにおいて出力する速さと入力する速さのズレを補うために、データを一時的に保持しておくためのメモリ上に確保された領域である。あらかじめ確保されていたバッファを上回る量のデータが入力された場合、データが入りきらず隣接する別のメモリに影響を与える可能性がある。これをバッファオーバフローという。

2　実例

ここで C 言語を用いてバッファオーバフローの実例を見ていく。

図 1 のコードではサイズ 5 の文字列型配列 a のバッファがすでに確保されており、また整数型変数 x に 10 が代入されている。

```c
// 学籍番号：4619072　氏名：服部　翼
#include <stdio.h>

int main(void)
{
    int x;
    char a[5];
    x = 10;

    printf("aを入力してください。\n");
    scanf("%s", a);

    printf("xの値は, %d\n", x);

    return 0;
}
```

図 1: SampleA のコード

```
[redacted]$ ./SampleA
aを入力してください。
aaaa
xの値は, 10
```

図 2: 入力例 1(a)

図 2 のように、配列 a に四文字までの文字列を入力すると null 文字も合わせてサイズ 5 に収まるため、通常通り x の値は 10 と表示されている。

```
[redacted]$ ./SampleA
aを入力してください。
aaaaa
xの値は，0
```

図 3: 入力例 1(b)

次に図 3 のように、五文字以上の文字列を入力してみるとここでバッファオーバフローが発生し、x の値を格納していたメモリが不正な値になり、正しく表示されなくなる。

```c
// 学籍番号：4619072　氏名：服部　翼
#include <stdio.h>

int main(void)
{
    int x;
    char a[5];
    x = 10;

    printf("x　　のアドレスは　%p\n", &x);
    printf("aの先頭のアドレスは　%p\n", a);
    printf("aを入力してください。");
    scanf("%s", a);

    printf("x　　のアドレスは　%p\n", &x);
    printf("aの先頭のアドレスは　%p\n", a);
    printf("xの値は，%d\n", x);

    return 0;
}
```

図 4: SampleB のコード

```
[redacted]$ ./SampleB
x　　のアドレスは　0x7ffee3467a98
aの先頭のアドレスは　0x7ffee3467a93
aを入力してください。aaaa
xの値は，10
```

図 5: 入力例 2(a)

図 4 のコードを実行した結果をみると、x のアドレスは配列 a の先頭のアドレスの 5byte 後ろにあることがわかる。図 5 の入力例 2(a) ではサイズ 1byte の文字 (char 型) が 4 つと null 文字を合わせて 5byte のデータが入力されており、あらかじめ確保していたバッファに収まる。

```
[redacted]$ ./SampleB
x　　のアドレスは　0x7ffee46efa98
aの先頭のアドレスは　0x7ffee46efa93
aを入力してください。aaaaa
xの値は，0
```

図 6: 入力例 2(b)

今度は図 6 の入力例 2(b) のように 5byte を超えるデータを入力すると配列 a のバッファを超えてしまい、後ろにある x のメモリに影響を及ぼしてしまうことがわかる。

以上のような仕組みでバッファオーバフローが発生する。これは非常に危険な脆弱性となるため、コードを書くときには入力データを厳密にチェックしたり、セキュリティ上で危険な関数などは使わないようにすることが重要である。