Laporan Penyisihan LKS SMK CYBER SECURITY 2021

P4\Ui
Reja Revaldy F.
Rezka Norhafizah

Contents

Penyisihan – Code Review

Code Review 1

Ringkasan soal

> Diberikan sebuah baris kode yang memilki kerentanan yang mana di sini kita disuruh untuk menemukan celah yang ada pada kode tersebut dan kami melakukan beserta patching (perbaikan) kodenya

Penyelesaian

1. Diberikan sebuah source code menggunakan Bahasa pemrograman C lalu kami coba buka source code nya dan menemukan function “gets “ yang dimana function ini rentan terhadap buffer overflow (ketika di compile maka akan diperingatkan bahwa function gets ini berbahaya).

```
warning: the `gets' function is dangerous and should not be used.
```

2. Lalu disini kami mencari alternative lain dari function gets dan mendapatkan function yang lebih aman yaitu “fgets”

```
┌──(kali㉿kali)-[~/lks/codereview]
└─$ cat code_1.c
#include <stdio.h>
#include <stdlib.h>

void echo(){
        printf("%s", "Enter a word to be echoed:\n"); char buf[128];
        fgets(buf, 128, stdin);
        printf("%s\n", buf);
}

int main() {
        echo();
}
```

LKSSMK2021{fe2aceacafda45ee549a4c6ded19fe1e}

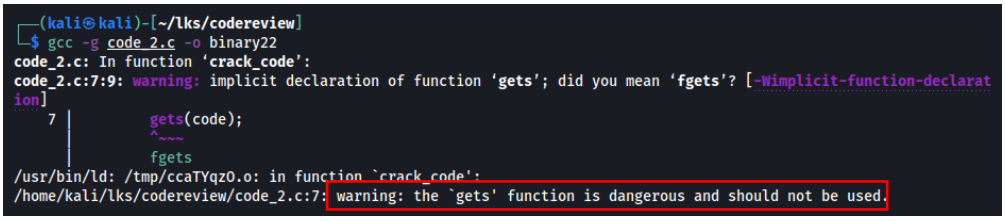
Code Review 2

Ringkasan soal

Diberikan sebuah baris kode yang memilki kerentanan yang mana di sini kita disuruh untuk menemukan celah yang ada pada kode tersebut dan kami melakukan beserta patching (perbaikan) kodenya

Penyelesaian

1. Untuk code review 2 juga hampir mirip dengan code review 1, disini kami menemukan function “gets”, lalu kami mencoba untuk mengganti nya dengan “fgets”. Kemudian ketika kami melakukan compile di source code awal maka akan diberikan peringatan untuk mengganti fungsi gets:



2. Lalu kami mencoba untuk melakukan patching dengan mengganti function “gets” ke “fgets” :

```
#include <stdio.h>

int crack_code() {
        char code[10];
        int val = 999, i;
        printf("Enter code: ");
        fgets(code, 10, stdin);
        for (i = 0; i < 10; i += 2) {

                val = (val & code[i]) | code[i + 1];
                val &= val >> code[i];
        }
        if (val == 101) {
                return 0;
        } else {
                return 1;
        }
}
void main() {
        if (crack_code()) {
                printf("Crack the secret code!\n");
        } else {
                printf("Now you know the secret!\n");
        }
}
```

LKSSMK2021{0b9812d6ce53646351b10c2ade68cec2}

Code Review 3

Ringkasan soal

Diberikan sebuah baris kode yang memilki kerentanan yang mana di sini kita disuruh untuk menemukan celah yang ada pada kode tersebut dan kami melakukan beserta patching (perbaikan) kodenya

Penyelesaian

1. Untuk code review ketiga kita diberikan source code php yang memiliki celah yaitu di bagian line 9 yang disisipkan kode <script> yang dimana bisa diserang dengan memasukan script, dan kami menemukan nama ancaman keamanan nya adalah “XSS Stored”

2. Kemudian kami melakukan patching seperti kode dibawah ini:

```php
<?php
if (isset($_POST['btnSign']))
{
    $message = trim($_POST['txtMessage']);
    $name = trim($_POST['txtName']);
    $message = strip_tags(addslashes($message));
    $message = mysqli_real_escape_string($GLOBALS["___mysqli_ston"], $message);
    $message = htmlspecialchars($message);
    $name = preg_replace('/<(.*)s(.*)c(.*)r(.*)i(.*)p(.*)t/i', '', $name);
    $name = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_stri
ng($GLOBALS["___mysqli_ston"], $name);

    $query = "INSERT INTO guestbook ( comment, name ) VALUES ( '$message', '$name' );";
    $result = mysqli_query($GLOBALS["___mysqli_ston"], $query) or die(1);
}
?>
```

3. Disini kami mengganti function “str_replace” di line 9 dengan function “preg_replace” kami mencari function alternative untuk function “str_replace” dan menambahkan patching di variable $name di line 10 menggunakan referensi yang kami temukan

LKSSMK2021{d93fafdb4f7924fb5f5bad6b4634767d}