Hope Foundation's
Finolex Academy of Management and Technology, Ratnagiri

Department of Information Technology

Subject name: Advance Security Lab			Subject Code: ITL702
Class	BE	Semester –VII	Academic year: 2019-20
Name of Student			**QUIZ Score :**
Roll No		Experiment No.	03
Title: **Implement and analyze Buffer Overflow attack.**			

1. Lab objectives applicable: LOB1, LOB3

2. Lab outcomes applicable: LO1, LO2

3. Learning Objectives:

1. To understand system vulnerabilities
2. To be alert about the bad programming practices.

4. Practical applications of the assignment/experiment: Industries/organizations create awareness about the bad programming practise used by their employees.

5. Prerequisites:

1. C programming.

6. Hardware Requirements:

A computer system with linux os

7. Software Requirements:

GCC compiler

8. Quiz Questions (if any): (Online Exam will be taken separately batch-wise, attach the certificate/ Marks obtained)

Q1. WHAT IS BUFFER OVERFLOW?

Q2. HOW TO OVERCOME BUFFER OVERFLOW VULNERABILITY?

9. Experiment/Assignment Evaluation:

Sr. No.	**Parameters**		**Marks obtained**	**Out of**
1	Technical Understanding (Assessment may be done based on Q & A **or** any other relevant method.) Teacher should mention the other method used -			6
2	Neatness/presentation			2
3	Punctuality			2
Date of performance (DOP)		**Total marks obtained**		**10**

Signature of the faculty

10. Theory:

What is Buffer Overflow?

A buffer is said to be overflow when the data gets written past the left or the right boundary of the buffer. This way the data gets written to a portion of memory which does not belong to the program variable that references the buffer.

Here is an example: char buff[10];

Here, buff' represents an array of 10 bytes where buff[0] is the left boundary and buff[9] is the right boundary of the buffer.

buff[10] = 'a';

In the above example, index 10 was used to store the value 'a'. This is the point where buffer overflow happens because data gets written beyond the right boundary of the buffer.

Buffer Overflow Attacks

We learned what buffer overflows is. But, that it is not the worst part. It gets worse when an attacker comes to know about a buffer over flow in your program and he exploits it. Following is the C program which is vulnerable to Buffer overflow attack.

11. Learning Outcomes Achieved

1. Understood the system vulnerabilities.

12. Conclusion:

1. **Applications of the studied technique in industry**
    a. Alert about the bad programming practise followed by the software engineers.
2. **Engineering Relevance**
    a. Helpful in designing a buffer overflow free software.
3. **Skills Developed**
    a. Good programming practise.

13. References:

[1]. https://www.geeksforgeeks.org/buffer-overflow-attack-with-example/

[2]. https://www.owasp.org/index.php/Buffer_Overflow