ACM Student Chapter

CTF (Capture The Flag) Challenge - 1

Buffer Overflow

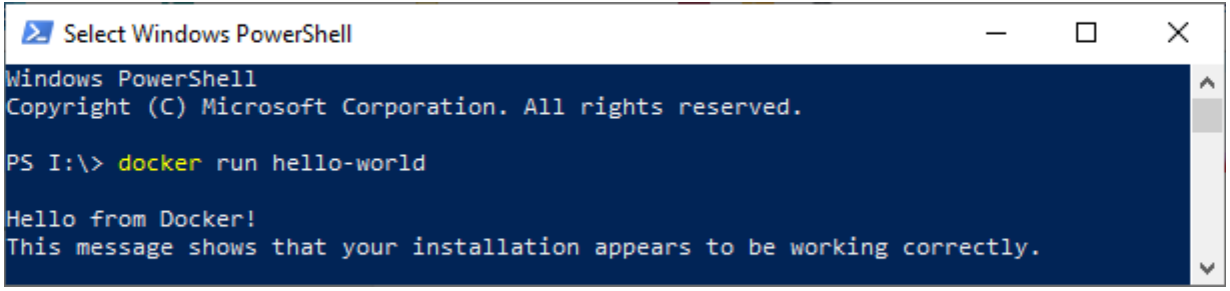
First, check the following video [pay more attention to the section that describes Buffer Overflow] https://www.youtube.com/embed/_GzE99AmAQU .

Initial Setup

Download and install Docker Community Edition (Important: check your OS).
https://store.docker.com/search?type=edition&offering=community
To test docker, open a shell CLI and type;

```
docker run hello-world
```

If the text “hello world” is displayed on your CLI, you can continue.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS I:\> docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.
```

LAB- 1

On a shell CLI type;

```
: ~ $  docker run  -it winwin/bovfl1
```

This command will download the bovfl1 image from the Docker hub called winwin and create a container from the image. The container is designed to run a program at the end of its creation. This program will prompt the user to enter a username and password. If you give username as admin and then give the appropriate password (which you do not know obviously) the program will print a security code (see the Image in the next page).
In this challenge your task is to hack this program and force it to print the admin’s security code. Fortunately, a part of the source code is available to you (see the box in next page- it may

be an old version of the program). However, in most cases you may have access to byte code only- not source. You can simply try buffer overflow attack to find the security code (but other attacks are also possible, for example brute-force attack or use reverse engineering, see https://yurichev.com/writings/RE4B-EN.pdf).

Consider the following algorithm.

```
Step 1: uType ← “”
Step 2: username ← “”
Step 3: password ← “”
Step 4:  Prompt and read username
Step 5:  Prompt and read password
Step 6:  if  (username == ”adminX” and password == “passwordX”)
                    Print “Secret Code X”
```

If the above algorithm is executed on **an ideal computing agent**, the output will never be “Secret Code X”, unless otherwise the user provides appropriate username and password in step 4 and step 5.

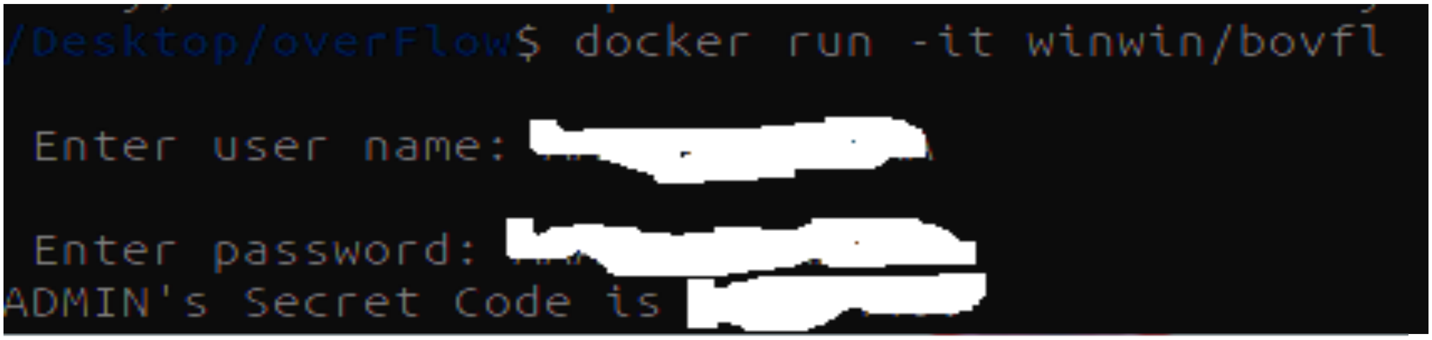
However, in a real situation the computing agent (consider a notional machine that includes CPU, RAM, OS and Compiler) may not be ideal- it may be vulnerable to buffer overflow attack.

In this lab, a container is created with a vulnerable program running on a vulnerable notional machine.

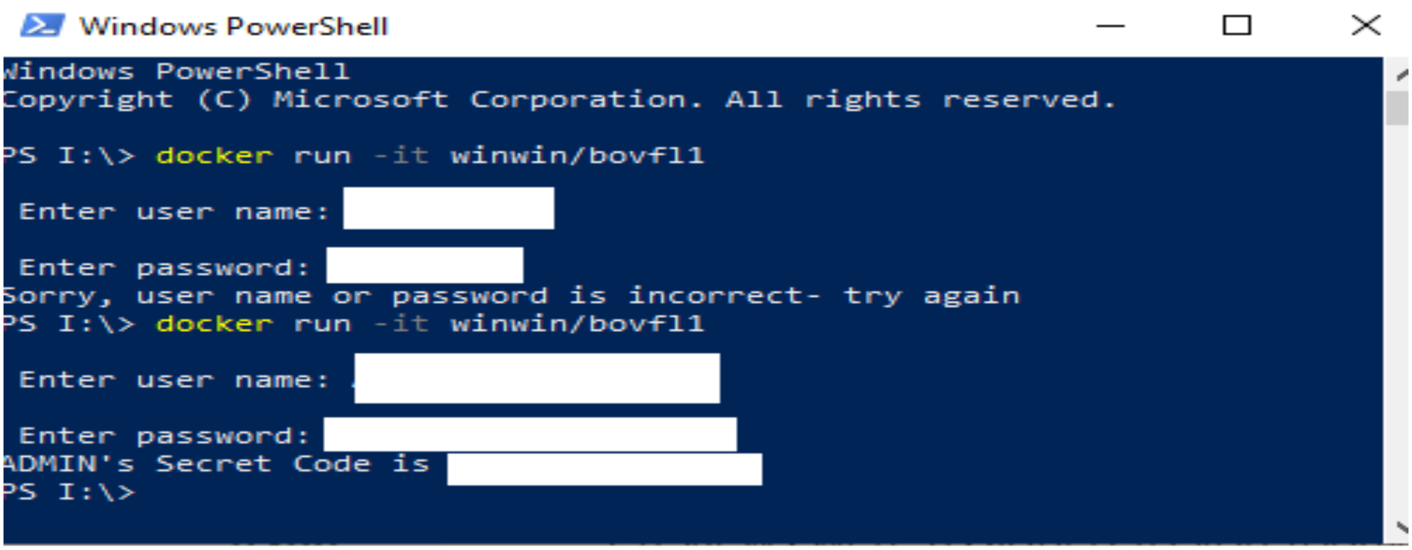
Your task is to design an attack vector to hack the program to reveal the secret code of the admin.

If you can successfully hack, send the Secret Code to the following email: mohanara@uncp.edu
A cash price will be awarded for the first person who send the correct security code.

```
#include <stdio.h>
#include <string.h>
#include <openssl/sha.h>

int main(){
  char uType = 'U'; // unknown user
  char uName[5];
  char pw[5];
  printf("\n Enter user name: ");
  scanf("%s", uName);
  printf("\n Enter password: ");
  scanf("%s", pw);
  if (uType='A')
     printf("ADMIN's Secret Code is ……………….
```



```
/Desktop/overFlow$ docker run -it winwin/bovfl

 Enter user name: [redacted]

 Enter password: [redacted]
ADMIN's Secret Code is [redacted]
```



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS I:\> docker run -it winwin/bovfl1

 Enter user name: [redacted]

 Enter password: [redacted]
Sorry, user name or password is incorrect- try again
PS I:\> docker run -it winwin/bovfl1

 Enter user name: [redacted]

 Enter password: [redacted]
ADMIN's Secret Code is [redacted]
PS I:\>
```

LAB- 2 (Advanced- for CS/IT students)

If you have completed LAB-1, in this lab, you are required to explain the attack vector you used.

Open a shell CLI and type;

```
: ~ $   docker run  -it winwin/bovfl1  bash
```

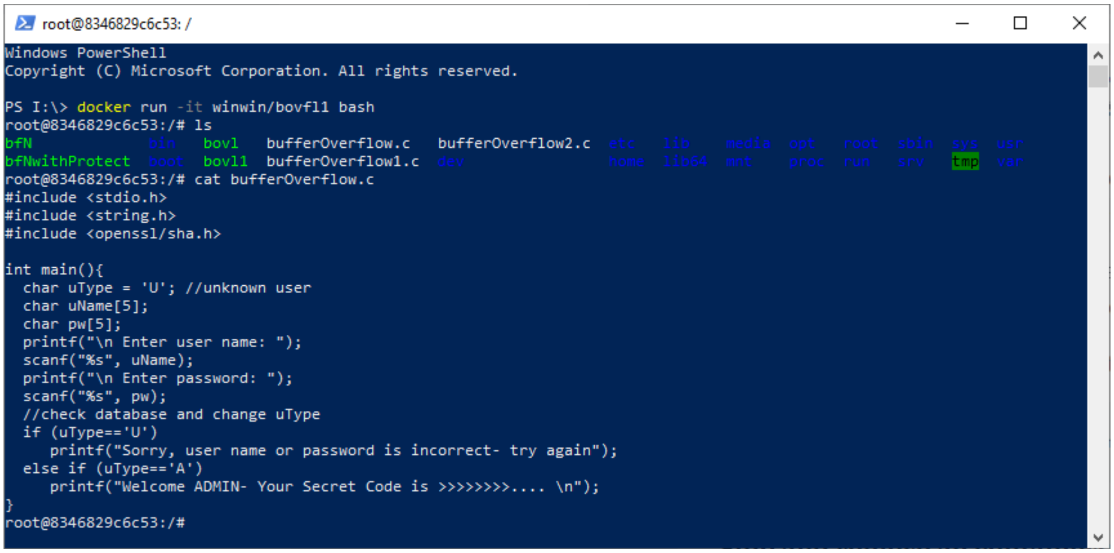
This command will create a container as in Lab-1, and also give interactive terminal to access the container. Now, you can inspect the contents of the container. The container includes a vulnerable OS and a C compiler. Some of the protection mechanisms are disabled. You can modify the programs or write your own programs and run it inside the container.

Your task in this assignment is explain your attack vector in LAB-1 using memory addresses of the variables.

```
:/# ls
```

There are three c programs (source code). To see the code use cat command.

```
:/# cat bufferOverflow.c
```



root@8346829c6c53: /

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS I:\> docker run -it winwin/bovfl1 bash
root@8346829c6c53:/# ls
bfN             bin   bovl   bufferOverflow.c   bufferOverflow2.c  etc   lib    media  opt   root  sbin  sys  usr
bfNwithProtect  boot  bovl1  bufferOverflow1.c  dev                home  lib64  mnt    proc  run   srv   tmp  var
root@8346829c6c53:/# cat bufferOverflow.c
#include <stdio.h>
#include <string.h>
#include <openssl/sha.h>

int main(){
  char uType = 'U'; //unknown user
  char uName[5];
  char pw[5];
  printf("\n Enter user name: ");
  scanf("%s", uName);
  printf("\n Enter password: ");
  scanf("%s", pw);
  //check database and change uType
  if (uType=='U')
     printf("Sorry, user name or password is incorrect- try again");
  else if (uType=='A')
     printf("Welcome ADMIN- Your Secret Code is >>>>>>>>.... \n");
}
root@8346829c6c53:/#
```

To edit you need to use nano command., or you may install your favourite editor using ‘apt-get install’ command.

```
:# nano bufferOverflow.c
```

The file bovl1 is the executable used in Lab-1, to try Lab-1 type.

```
:# ./bovl1
```

The secret code is removed from all the source codes in this container. But, bovl1 is the compiled code of bufferOverflow1.c with the secret code (with disabled protections). if you can successfully hack, the secret code will be displayed.

You can edit the given source code and compile and run the program in the container.

To compile bufferOverflow.c, for example, try

```
:# gcc bufferOverflow.c -o bfNtest1
```

Now check the folder to see bfNtest1 exists. To execute type

```
:# .\bfNtest1
```

Note that, bfNtest1 will be protected against stack smasing.

Check the first comment included in bufferOverflow1.c to disable this compiler level protection using the directive ‘-fno-stack-protector’.
The file bufferOverflow2.c includes some help on displaying the memory addresses of the variables.
Your task is to construct a simple attack vecor to hack the program bovl1, and explain how it will work based on the memory locations of the variables.
GOOD LUCK!

UNCP Store
Not secure | 10.24.29.50/bookstore/
BOOKSTORE
Shop
Textbooks
Search Keywords or ISBN
Store Info
Sign in
Bag
MOM
SUPERHERO. WARRIOR. CONFIDANT. BEST FRIEND.
Shop Now ›
Type here to search
71°F Sunny
4:00 PM
5/9/2022

UNCP- Bookstore

Congratulations
Enter your Promocode

Give the Promocode | Submit

Show □

Shop Categories

Men's T-Shirts & Tanks →

Apple →

Women's T-Shirts & Tanks →

Drinkware →

Auto Accessories →

Diploma Frames →

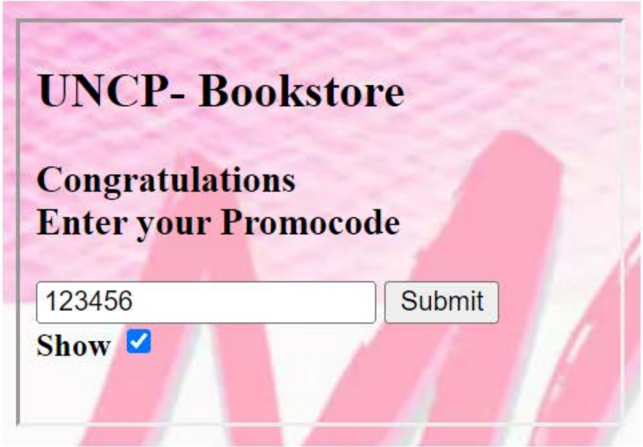
Face Masks & Covers →

Earbuds & Headphones →

School Accessories →

Blankets →

"123456" Promo Code Example

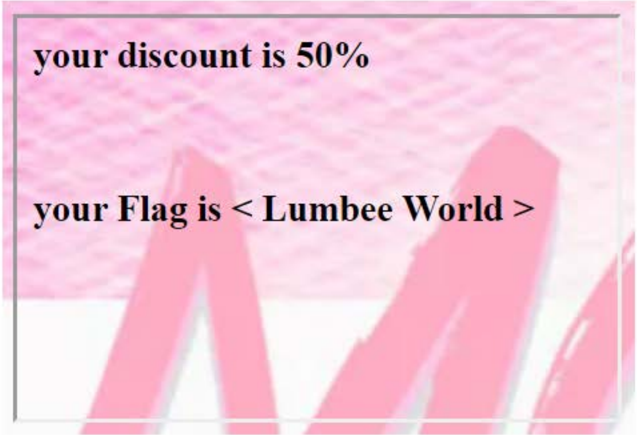


UNCP- Bookstore

Congratulations
Enter your Promocode

123456

Show ☑



your discount is 50%

your Flag is < Lumbee World >

Subject:	ACM Challenge $100 UNCP Bookstore Gift Card
Date:	Monday, May 9, 2022 at 4:18:19 PM Eastern Daylight Time
From:	Jean Choi
To:	Selvarajah Mohanarajah, Prashanth BusiReddyGari, Jessica Conner-Strunk, Elliott Hollifield, Nicky Bullard, Mariana Yanez-Diaz, Sereena Chavis, John Matthew O Bebonia, Joel Davis, Gerald Drye, Matthew S Edwards, Demi S Meiklejohn, Joseph Smith-js0099, Caleb Gilbert, Ernest Brown, Jose Garcia Vergara, Eric Savage, Roman Watson, Marcus Robinson, Ashabori Mayurakkhi, Alexander Summerlot, Robert Huynh, James Shover, Raylond Gilbert, Anthony Vazquez, Richard Kiyingi, Kyle Gause, Joshua Owens, Dylan Sikinger
Attachments:	Bookstore_SS.png, 123456_example.png

ACM Members,

First, I'd like to wish everyone good luck during this finals week! Also, big congratulations to those who are graduating this spring- Marianna Yanez-Diaz and Sereena Chavis.

ACM Challenge:
Using a computer in the SCI1202 lab, access the local bookstore's website:
uncp1.edu/bookstore
or
10.24.29.50/bookstore
(you can only access these sites from the SCI1202 lab).

The website asks for a valid promo code to receive a gift card. I've attached a screenshot of what happens when a valid promocode such as "123456" is entered. Since this is the example that Mohan showed us two meetings ago, it is not valid to be used for this competition. For this challenge, users must attempt to hack the site to receive a gift card. The first person to send me a screenshot of successfully hacking the site will receive a $100 UNCP Bookstore gift card. This challenge will close this Friday, May 13th at 11:59 PM.

Best,
Jean Choi