2022 CAT-CERT 스터디 보고서

컴퓨터정보공학부 202221168 김지혜

2022.05.20#System\_study-4

실습 #1

소스코드

|  |
| --- |
| .global main  main:  push %ebp  mov %esp, %ebp  sub $0x100, %esp  mov $0x03, %eax  mov $0x00, %ebx  lea -0x100(%ebp), %ecx  mov $0x70, %edx  int $0x80    xor %eax, %eax  xor %ebx, %ebx    mov $0x04, %eax  mov $0x01, %ebx  int $0x80  mov $0x1, %eax  int $0x80 |

출력 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

실습#2

소스코드

|  |
| --- |
| .global main  main:  push %ebp  mov %esp, %ebp  sub $0x100, %esp    mov $0x05, %eax  mov $File, %ebx  mov $0x00, %ecx  mov $0644, %edx  int $0x80  mov $0x03, %eax  mov $0x03, %ebx  lea -0x100(%ebp), %ecx  mov $0x18, %edx  int $0x80    mov $0x04, %eax  mov $0x01, %ebx  int $0x80    mov $0x01, %eax  int $0x80  .data  File:  .string "AAAA.txt" |

출력결과

텍스트이(가) 표시된 사진

자동 생성된 설명

실습#3

테이블이(가) 표시된 사진

자동 생성된 설명

‘eax’는 32비트인 레지스터이고 쉘코드를 널없이 만들기 위해서는 앞에 2개만 리틀레디안 방식으로 바이트 단위로 뒤에서부터 작성된다. 쉘코드를 널없이 만들기 위해서 뒤에 1바이트만 필요하므로 eax를 al로 고친다

Prob8

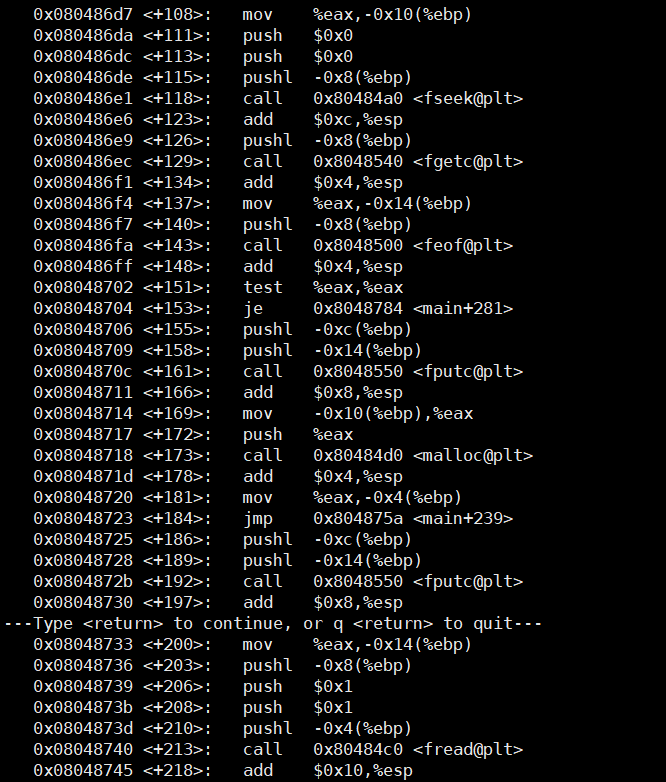
소스코드

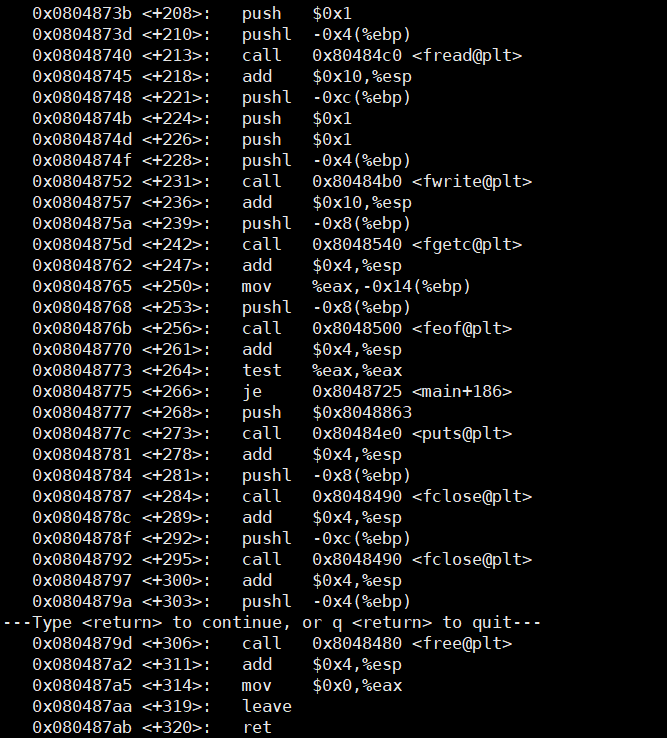
|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  int main(int argc, char\*argv[]){  int\*a;  if(argc<=2){  puts("Usage:./prob8 input\_file.txt output\_file.txt");  exit(1);  }  else{  FILE\*b=fopen(argv[1],"r+");  FILE\*c=fopen(argv[2],"w+");    fseek(b,0,2);  int d=ftell(b);    fseek(b,0,0);  int e=fgetc(b);//14  if(feof(b)){//281  fputc(e,c);  a=(int\*)malloc(d);//239  while((e=fgetc(b)),!feof(b)){  e=fputc(e,c);  fread(a,1,1,b);  fwrite(a,1,1,c);  }    puts("gogo~!");  }  fclose(b);  fclose(c);  free(a);  return 0;    }  } |

출력결과

텍스트이(가) 표시된 사진

자동 생성된 설명



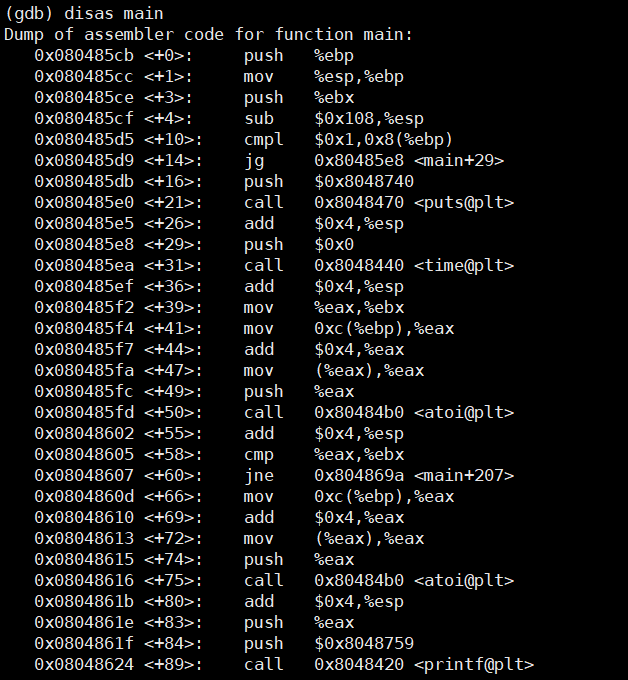


Prob9

소스코드

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  #include<string.h>  #include<time.h>  int main(int argc,char\*argv[]){  FILE\*a;  int b;  int \*c;  char arr[256];  if(argc<=1){  puts("Usage : ./prob9 [passwd]");  }  if(time(0)==atoi(argv[1])){  printf("passwd = %d\n",atoi(argv[1]));  a=fopen("key\_09.txt","r");  fscanf(a,"%s",arr);  c=(int\*)malloc(strlen(arr)+1);  strcpy(c,arr);  printf("key = %s",c);    }  else{  puts("Try Again~~!");  }  } |

출력결과



텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

#파일을 쓰고 그 파일을 다시 읽는 프로그램 만들기

소스코드

|  |
| --- |
| .global main  main:  push %ebp  mov %esp, %ebp  sub $0x70, %esp  mov $0x05, %eax  mov %esp, %ebx  mov $file, %ebx  mov $0x02, %ecx  int $0x80  mov $0x03, %eax  mov $0x00, %ebx  mov %esp, %ecx  mov $0x18, %edx  int $0x80  mov $0x04, %eax  mov $0x03, %ebx  int $0x80  //파일읽고출력하기  mov $0x03,%eax  mov $0x03, %ebx  lea -0x70(%ebp), %ecx  mov $0x18, %edx  int $0x80  mov $0x04, %eax  mov $0x01, %ebx  int $0x80  mov $0x01, %eax  int $0x80  .data  file:  .string "touch.txt" |

출력결과

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명

#Shellcode에 getuid setreuid 추가해서 만들기

소스코드

|  |
| --- |
| .global main  main:  push %ebp  xor %eax, %eax  xor %ebx, %ebx  xor %ecx, %ecx  xor %edx, %edx  mov $0xb, %al  push %edx  push $0x68732f2f  push $0x6e69622f  mov %esp, %ebx  push %edx  mov %esp, %ecx  int $0x80  xor %eax, %eax  mov $0x18, %eax  int $0x80  mov $0x46, %eax  mov $1117, %ebx  mov $1117, %ecx  int $0x80 |

출력결과

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명