

Solution 5

Array

Consider the following declaration:

```
short S[5][6];  
int T[3][4];  
float U[1][2];  
char *V[2][0];  
char *W[2][3][4];
```

And fill in the following blanks:

Array	Size of Element	Size of Arrsy	Start Address	Element	ElementAddress
S	2	60	s	S[i][j]	$s+2*(i*6+j)$
T	4	48	t	T[i][j]	$t+4*(i*4+j)$
U	4	8	u	U[i][j]	$u+4*(i*2+j)$
V	4	0	v	/	/
W	4	96	w	W[i][j][k]	$w+4*(j*12+j*4+k)$

Data Structure

```
struct data {  
    char a;  
    short b[2];  
    char* c;  
    union  
    {  
        char x;  
        short y;  
        int z;  
    }p;  
    char d;  
};
```

```
struct data d[2];
```

Suppose the address of global variable dis 0x8049600, please answer the following questions.

Variable	Start address
d[0]	0x8049600
d[1]	[1] 0x8049614
d[0].a	[2] 0x8049600
d[0].b[1]	[3] 0x8049604
d[0].c	[4] 0x8049608
d[0].p.y	[5] 0x804960C
d[0].p.z	[6] 0x804960C
d[0].d	[7] 0x8049610

Pointers

What's the output of the following C program? (on a 32-bit machine)

```
int main()
{
    static char char_table[3][13] = {
        { 'd', 'o', 32, 'y', 'o', 'u', 32, 'w', 'a', 'n', 't', 32, 'a' },
        { 32, 109, 105, 100, 116, 101, 114, 109, 32, 101, 120, 97, 109 },
        { 0 }
    };
    static char ans[] = "abcdefghijklmnopqrstuvwxyyz";
    printf("%s?\n", char_table);
    printf("%c%c%c!\n",
        (char) (((char **)ans)[6]),
        (char) (((char *)ans)[4]),
        (char) (ans[18]));
    return 0;
}
```

Answer:

do you want a midterm exam?

yes!

Buffer Overflow

Suppose we have the following function 'login' to perform login process.

```
int login()
{
    char username[8];
```

```
char password[8];

gets(username);
gets(password);
return check_match_in_database(username, password);
}
```

Here is a part of the function's assembly.

```
pushl %ebp
movl %esp, %ebp
subl $40, %esp
leal -16(%ebp), %eax
movl %eax, (%esp)
call _gets
leal -24(%ebp), %eax
movl %eax, (%esp)
call _gets
.....
```

In the normal process, if the username and the password are both ok, the function 'login_ok' will be called to indicate login success. We've already known that the address of 'login_ok' is 0x804013da. Can you construct an input to make the function 'login_ok' be called after 'login' returns? You need to specify the key bytes and their positions rather than the complete input. And give one brief explanation about your input.

Answer:

得分点:

1. 需要提到在 username+20Bytes 的地方 填入 0x804013da

也就是

```
username[20] = 0xda;
username[21] = 0x13;
username[22] = 0x40;
username[23] = 0x80;
```

2. 提到这个地址是为了覆盖 login 函数调用前被压入的 return address。