리버스 엔지니어링 바이블

ch06. 흔히 사용하는 패턴

저자 : 강병탁

김영철

2015. 12. 16.

문자열 컨트롤

- strcpy 함수

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

void main()
{
        char *source = "Hello world";
        char target[100];

        strcpy(target, source);
        printf("%s\n", target);
}
```

문자열 컨트롤

- strcpy 함수

```
push    ebp
mov     ebp, esp
sub     esp, 13Ch
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_13C]
mov     ecx, 4Fh
mov     eax, 0CCCCCCCCh
rep stosd
mov     eax, dword_417000
xor     eax, ebp
mov     [ebp+var_4], eax
mov     [ebp+Source], offset aHelloWorld ; "Hello world"
mov     eax, [ebp+Source]
push    eax             ; Source
lea     ecx, [ebp+Dest]
push    ecx             ; Dest
call    j_strcpy
add     esp, 8
mov     esi, esp
lea     eax, [ebp+Dest]
push    eax
push    offset Format   ; "%s₩n"
call    ds:printf
```

```
push    ebp
mov     ebp, esp
and     esp, 0FFFFFFF0h
add     esp, 0FFFFFF80h
mov     eax, large gs:14h
mov     [esp+7Ch], eax
xor     eax, eax
mov     dword ptr [esp+14h], offset aHelloWorld ; "Hello world"
mov     eax, [esp+14h]
mov     [esp+4], eax    ; src
lea     eax, [esp+18h]
mov     [esp], eax      ; dest
call    _strcpy
lea     eax, [esp+18h]
mov     [esp], eax      ; s
call    _puts
```

문자열 컨트롤

- strcat 함수

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

int main()
{
        char path[100];

        strcat(path, "strcat");
        printf("%s\n", path);
        return 0;
}
```

문자열 컨트롤

- strcat 함수

```
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_130]
mov     ecx, 4Ch
mov     eax, 0CCCCCCCCh
rep stosd
mov     eax, dword_417000
xor     eax, ebp
mov     [ebp+var_4], eax
push    offset Source   ; "strcat"
lea     eax, [ebp+Dest]
push    eax             ; Dest
call    j_strcat
```

```
xor     eax, eax
lea     eax, [esp+18h]
mov     ecx, 0FFFFFFFFh
mov     edx, eax
mov     eax, 0
mov     edi, edx
repne scasb
mov     eax, ecx
not     eax
lea     edx, [eax-1]
lea     eax, [esp+18h]
add     eax, edx
mov     dword ptr [eax], 63727473h
mov     word ptr [eax+4], 7461h
mov     byte ptr [eax+6], 0
```

문자열 컨트롤

- strlwr 함수

```
#include <string.h>

int main()
{
        char target[] = "ABcDe";
        strlwr(target);
        printf("%s\n", target);
        return 0;
}
```

문자열 컨트롤

- strlwr 함수

```
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_D4]
mov     ecx, 35h
mov     eax, 0CCCCCCCCh
rep stosd
mov     eax, dword_417000
xor     eax, ebp
mov     [ebp+var_4], eax
mov     eax, ds:dword_415740
mov     dword ptr [ebp+Str], eax
mov     cx, ds:word_415744
mov     [ebp+var_C], cx
mov     esi, esp
lea     eax, [ebp+Str]
push    eax             ; Str
call    ds:_strlwr
```

문자열 컨트롤

- strcmp 함수

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

int main()
{
        char *target1="hello";
        char *target2="hello";

        printf("%d\n", strcmp(target1, target2));

        return 0;
}
```

문자열 컨트롤

- strcmp 함수

```
push    ebp
mov     ebp, esp
sub     esp, 0D8h
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_D8]
mov     ecx, 36h
mov     eax, 0CCCCCCCCh
rep stosd
mov     [ebp+Str1], offset aHello ; "hello"
mov     [ebp+Str2], offset aHello ; "hello"
mov     eax, [ebp+Str2]
push    eax             ; Str2
mov     ecx, [ebp+Str1]
push    ecx             ; Str1
call    j_strcmp
```

```
push    ebp
mov     ebp, esp
and     esp, 0FFFFFFF0h
sub     esp, 20h
mov     dword ptr [esp+18h], offset aHello ; "hello"
mov     dword ptr [esp+1Ch], offset aHello ; "hello"
mov     eax, [esp+1Ch]
mov     [esp+4], eax    ; s2
mov     eax, [esp+18h]
mov     [esp], eax      ; s1
call    _strcmp
```

문자열 컨트롤

- strlen 함수

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

int main()
{
        char *target="strlen function";

        printf("%d\n", strlen(target));
        return 0;
}
```

문자열 컨트롤

- strlen 함수

```
push    ebp
mov     ebp, esp
sub     esp, 0CCh
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_CC]
mov     ecx, 33h
mov     eax, 0CCCCCCCCh
rep stosd
mov     [ebp+target], offset aStrlenFunction ; "strlen function"
mov     eax, [ebp+target]
push    eax             ; Str
call    j__strlen
```

```
push    ebp
mov     ebp, esp
and     esp, 0FFFFFFF0h
sub     esp, 20h
mov     dword ptr [esp+1Ch], offset aStrlenFunction ; "strlen function"
mov     eax, [esp+1Ch]
mov     [esp], eax      ; s
call    _strlen
```

문자열 컨트롤

- strtol 함수

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

int main()
{
        char *target = "2015hello";
        char *nan;

        printf("%ld\n", strtol(target, &nan, 10));
        return 0;
}
```

문자열 컨트롤

- strtol 함수

```
push    ebp
mov     ebp, esp
sub     esp, 0D8h
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_D8]
mov     ecx, 36h
mov     eax, 0CCCCCCCCh
rep stosd
mov     [ebp+target], offset a2015hello ; "2015hello"
mov     esi, esp
push    0Ah             ; Radix
lea     eax, [ebp+nan]
push    eax             ; EndPtr
mov     ecx, [ebp+target]
push    ecx             ; Str
call    ds:__imp__strtol
```

```
push    ebp
mov     ebp, esp
and     esp, 0FFFFFFF0h
sub     esp, 20h
mov     dword ptr [esp+1Ch], offset a2015hello ; "2015hello"
mov     dword ptr [esp+8], 0Ah ; base
lea     eax, [esp+18h]
mov     [esp+4], eax    ; endptr
mov     eax, [esp+1Ch]
mov     [esp], eax      ; nptr
call    _strtol
```

문자열 컨트롤

- strstr 함수

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

int main()
{
        char *target="hello world";

        printf("%s\n", strstr(target, "w"));
        return 0;
}
```

문자열 컨트롤

- strstr 함수

```
push    ebp
mov     ebp, esp
sub     esp, 0CCh
push    ebx
push    esi
push    edi
lea     edi, [ebp+var_CC]
mov     ecx, 33h
mov     eax, 0CCCCCCCCh
rep stosd
mov     [ebp+target], offset aHelloWorld ; "hello world"
push    offset _SubStr  ; "w"
mov     eax, [ebp+target]
push    eax             ; _Str
call    j_?strstr@@YAPADPADPBD@Z ; strstr(char *,char const *)
```

```
push    ebp
mov     ebp, esp
and     esp, 0FFFFFFF0h
sub     esp, 20h
mov     dword ptr [esp+1Ch], offset aHelloWorld ; "hello world"
mov     dword ptr [esp+4], 77h ; c
mov     eax, [esp+1Ch]
mov     [esp], eax      ; s
call    _strchr
```