binScan Design Document

A binary scanning utility

By Mark McCloskey

Table of Contents

Software Components

The software consists of 8 main components loosely broken into their respective functions. The components are C files with their corresponding header files and include: commandLine.c, disasm.c, entropy.c, err.c, main.c, md5.c, parseElf.c, and finally an assembly file unpass.asm.

commandLine.c

commandLine.c primarily focuses on managing input and output from the user.

disasm.c

disasm.c contains the code used to disassemble binaries.

entropy.c

entropy.c calculates the entropy of the binary file.

err.c

err.c is a simple error handling function.

md5.c

md5.c calculates the md5 hash of the .text section of the binary.

parseElf.c

parseElf.c is the workhorse of the group and handles parsing and delegating tasks related to finding information about the ELF file.

unpass.asm

unpass.asm is an assembly function written to be used in binScan.

main.c

main.c is the driver of binScan and delegates work to all of the above components.

Application Storage Protocol

The storage protocol in the application is a structure named ElfDetails.

```
typedef struct elfDetails {
        uint64_t sizeOfTextSection;
        void  *textData;
        unsigned char *md5Hash;
        uint64_t numDlopenCalls;
        double entropy;
        char *strings[NUM_STRING_ADDRS];
} ElfDetails;
```

Storage Protocol

This structure is passed around and populated during analysis of a binary and the information is saved to disk in much the same form.

Build Instructions

Navigate to the /bin folder and run ‘make’, after the process completes the binScan executable will be in the directory.

```
markdm@enee459b-1:~/project1/bin$ ls
Makefile
markdm@enee459b-1:~/project1/bin$ make
markdm@enee459b-1:~/project1/bin$ ls
binScan  Makefile
markdm@enee459b-1:~/project1/bin$
```

Building binScan

Requirements Mapping

	Requirement	File	Function	Line#
1	Application should be able to analyze ELF binaries	parseElf.c	All	24
2	Verify binary follows ELF format	parseElf.c	parseElf	68
3	Collect 5 classes of information	parseElf.c	parseElf	94
4	One attribute must be size of .text section	parseElf.c	parseElf	95
5	One attribute must be MD5 hash of .text section	md5.c	hash	11
6	One attribute must be entropy of file	entropy.c	calculateEntropy	10
7	One attribute must be # dlopen calls	disasm.c	countDlopens	54
8	Software shall use original binary format on disk	parseElf.h	saveFile	14
9	Software shall obfuscate contents on disk	parseElf.c	fuzzFile	298
10	Software shall provide an authentication mechanism	commandLine.c	getUsername/ getPassword	46
11	Store info in persistent manner	parseElf.c	saveFile	306
12	Software shall run on linux			
13	Software shall use libelf	parseElf.h		1
14	Software shall use openssl lib for MD5	md5.c		2
15	Software shall use capstone for disassembly	disasm.h	disasm	1
16	Software contains one intentional software vulnerability	unpass.asm	check	7
17	Software shall contain one function written in assembly	unpass.asm	Check	1
18	All function shall be written in C or ASM	They are		
19	Software shall be built and run on Linux for 32 bit with gcc.	Makefile		1
20	The software must compile and run on class VM’s	It does.		
O1	Resolve strings passed to dlopen	parseElf.c	parseElf	186