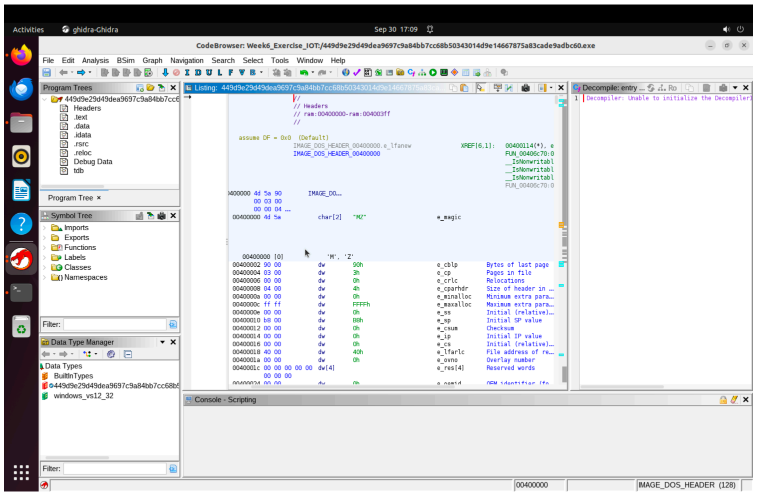
In_Class_Exercise_Week6

Lakshminath Reddy Alamuru
SUID: 367982169

Part1:
Set up the ghidra in linux machine of my virtual box, and analyzed the malware file that is provided and shared the screenshots here.

Part2:
Opened the malware for analysis, the entry point is 00400000.



CodeBrowser: Week6_Exercise_IOT:/449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a83cade9adbc60.exe

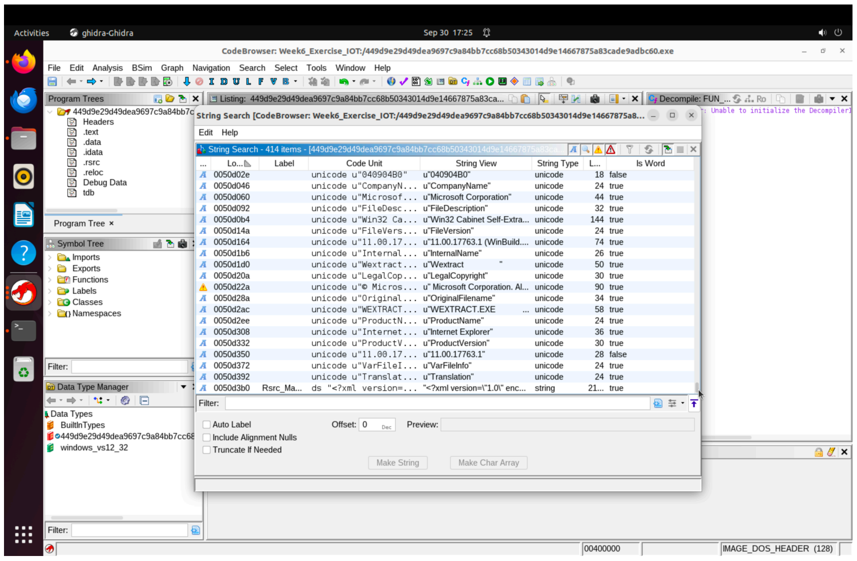
```
                     //
                     // Headers
                     // ram:00400000-ram:004003ff
                     //
  assume DF = 0x0  (Default)
                     IMAGE_DOS_HEADER_00400000.e_lfanew          XREF[6,1]:   00400114(*), e
                     IMAGE_DOS_HEADER_00400000                                FUN_00406c70:0
                                                                              __IsNonwritabl
                                                                              __IsNonwritabl
                                                                              __IsNonwritabl
                                                                              FUN_00406c70:0
00400000 4d 5a 90        IMAGE_DO...
         00 03 00
         00 00 04 ...
  00400000 4d 5a          char[2]    "MZ"                 e_magic

    00400000 [0]           'M', 'Z'
  00400002 90 00          dw         90h                  e_cblp       Bytes of last page
  00400004 03 00          dw         3h                   e_cp         Pages in file
  00400006 00 00          dw         0h                   e_crlc       Relocations
  00400008 04 00          dw         4h                   e_cparhdr    Size of header in ...
  0040000a 00 00          dw         0h                   e_minalloc   Minimum extra para...
  0040000c ff ff          dw         FFFFh                e_maxalloc   Maximum extra para...
  0040000e 00 00          dw         0h                   e_ss         Initial (relative)...
  00400010 b8 00          dw         B8h                  e_sp         Initial SP value
  00400012 00 00          dw         0h                   e_csum       Checksum
  00400014 00 00          dw         0h                   e_ip         Initial IP value
  00400016 00 00          dw         0h                   e_cs         Initial (relative)...
  00400018 40 00          dw         40h                  e_lfarlc     File address of re...
  0040001a 00 00          dw         0h                   e_ovno       Overlay number
  0040001c 00 00 00 00 00 dw[4]                           e_res[4]     Reserved words
           00 00 00
```

Decompiler: Unable to initialize the Decompiler

00400000 IMAGE_DOS_HEADER (128)

Hardcoded IP address or domains screenshots are mentioned below:
It seems not hardcoded domains are mentioned in the strings.



String Search [CodeBrowser: Week6_Exercise_IOT:/449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a8...

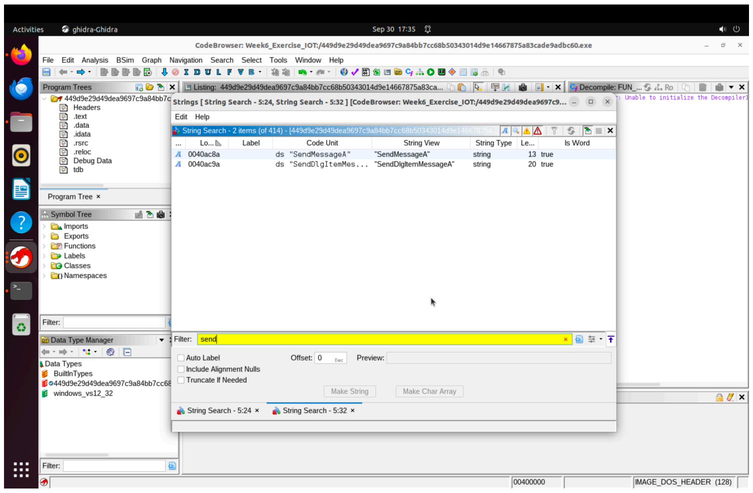
String Search - 414 items - [449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a83ca...

...	Lo...	Label	Code Unit	String View	String Type	L...	Is Word
A	0050d02e		unicode u"040904B0"	u"040904B0"	unicode	18	false
A	0050d046		unicode u"CompanyN...	u"CompanyName"	unicode	24	true
A	0050d060		unicode u"Microsof...	u"Microsoft Corporation"	unicode	44	true
A	0050d092		unicode u"FileDesc...	u"FileDescription"	unicode	32	true
A	0050d0b4		unicode u"Win32 Ca...	u"Win32 Cabinet Self-Extra...	unicode	144	true
A	0050d14a		unicode u"FileVers...	u"FileVersion"	unicode	24	true
A	0050d164		unicode u"11.00.17...	u"11.00.17763.1 (WinBuild....	unicode	74	true
A	0050d1b6		unicode u"Internal...	u"InternalName"	unicode	26	true
A	0050d1d0		unicode u"Wextract...	u"Wextract "	unicode	50	true
A	0050d20a		unicode u"LegalCop...	u"LegalCopyright"	unicode	30	true
⚠	0050d22a		unicode u"© Micros...	u" Microsoft Corporation. Al...	unicode	90	true
A	0050d28a		unicode u"Original...	u"OriginalFilename"	unicode	34	true
A	0050d2ac		unicode u"WEXTRACT...	u"WEXTRACT.EXE ...	unicode	58	true
A	0050d2ee		unicode u"ProductN...	u"ProductName"	unicode	24	true
A	0050d308		unicode u"Internet...	u"Internet Explorer"	unicode	36	true
A	0050d332		unicode u"ProductV...	u"ProductVersion"	unicode	30	true
A	0050d350		unicode u"11.00.17...	u"11.00.17763.1"	unicode	28	false
A	0050d372		unicode u"VarFileI...	u"VarFileInfo"	unicode	24	true
A	0050d392		unicode u"Translat...	u"Translation"	unicode	24	true
A	0050d3b0	Rsrc_Ma...	ds "<?xml version=...	"<?xml version=\"1.0\" enc...	string	21...	true

Auto Label
Include Alignment Nulls
Truncate If Needed
Offset: 0 Dec

00400000 IMAGE_DOS_HEADER (128)

Strings screenshot is mentioned below, DDOS is searched in strings and provided screenshot as well.



CodeBrowser: Week6_Exercise_IOT:/449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a83cade9adbc60.exe

Listing: 449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a83ca...

Decompile: FUN_...

Unable to initialize the Decompiler

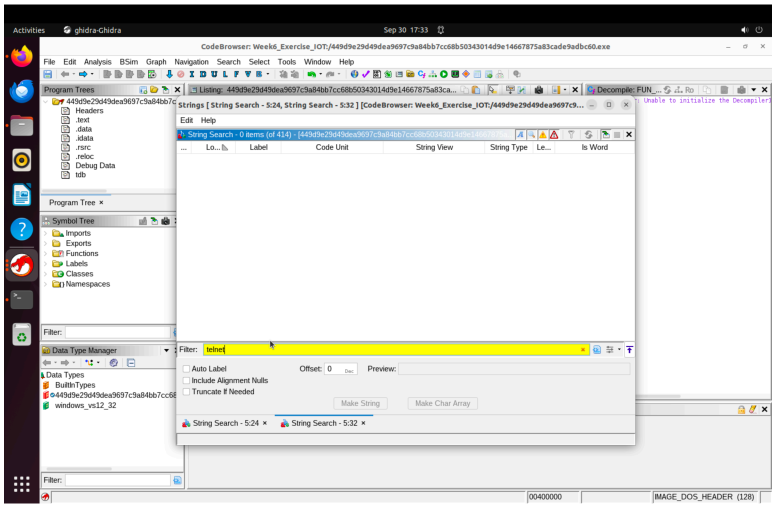
Strings [String Search - 5:24, String Search - 5:32] [CodeBrowser: Week6_Exercise_IOT:/449d9e29d49dea9697c9...

String Search - 2 items (of 414) - [449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a...

...	Lo...	Label	Code Unit	String View	String Type	Le...	Is Word
	0040ac8a		ds "SendMessageA"	"SendMessageA"	string	13	true
	0040ac9a		ds "SendDlgItemMes...	"SendDlgItemMessageA"	string	20	true

Filter: send

Screenshot fot telnet:



CodeBrowser: Week6_Exercise_IOT:/449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a83cade9adbc60.exe

Listing: 449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a83ca...

Decompile: FUN_...

Unable to initialize the Decompiler

Strings [String Search - 5:24, String Search - 5:32] [CodeBrowser: Week6_Exercise_IOT:/449d9e29d49dea9697c9...

String Search - 0 items (of 414) - [449d9e29d49dea9697c9a84bb7cc68b50343014d9e14667875a...

...	Lo...	Label	Code Unit	String View	String Type	Le...	Is Word

Filter: telnet

Part 3:

1. IoT malware targets many CPU architectures and custom libc/firmware so binaries are often cross-compiled and stripped, making static analysis harder.They're also highly network/firmware-focused (protocol quirks, persistence in flash), so behavior often only appears at runtime or on the device.
2. Hardcoded IPs/domains, wget/curl/busybox/sh strings and Xrefs to socket/connect/sendto or execve were red flags. Tight loops, thread creation for repeated sends, and nearby decode routines (XOR/ROL) strongly suggested DDoS/C2 activity. But it seems I don't find any the strings folder

3. Defined Strings + Xrefs and the Decompiler let you quickly trace where network/command strings are used; searching for 32-bit constants found raw IP encodings.

Scripting (Python/Jython) to bulk-grep strings/refs and using the Symbol Tree/Imports to locate networking/syscall callers sped up triage greatly.

4. Packed/obfuscated or stripped binaries and custom syscall wrappers hid intent and required finding decoder stubs or emulating behavior. Static analysis missed runtime-only behavior (dynamic C2, unpacking), so safe sandboxed execution or instrumentation was needed to confirm network/flooding actions.