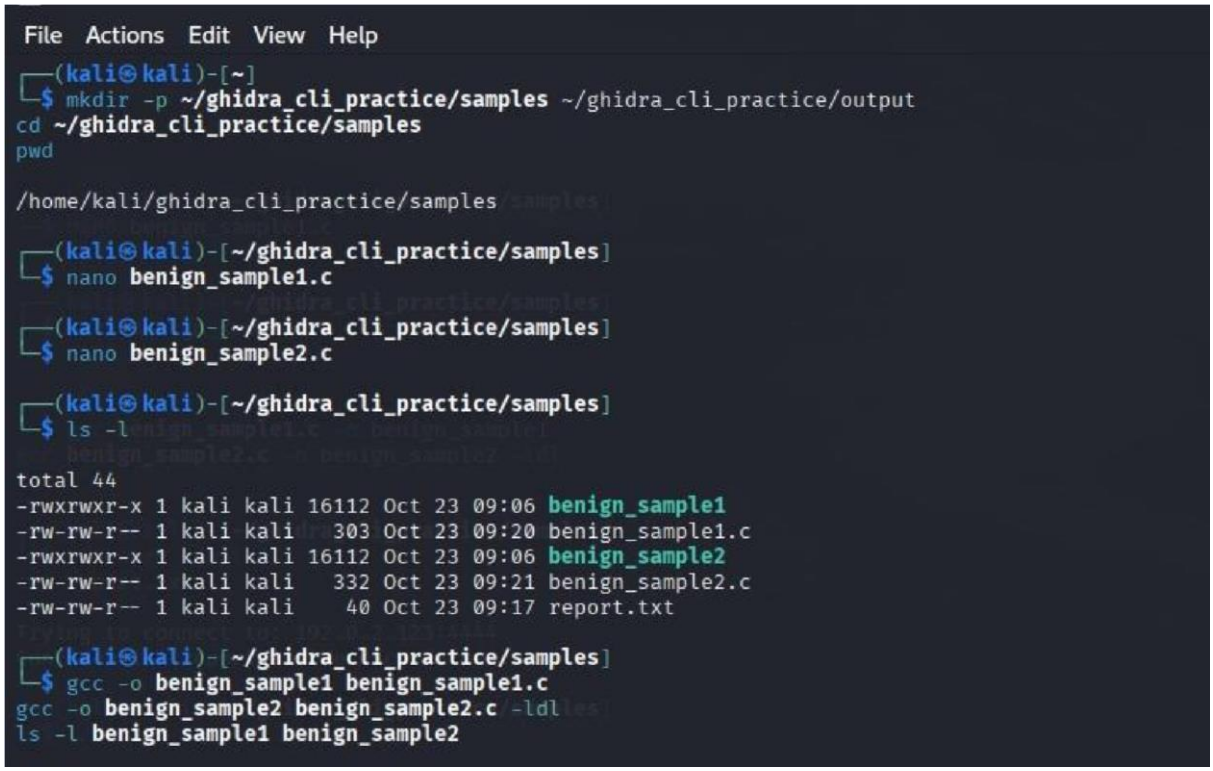
Ex No-10 : Use Ghidra to disassemble and analyze the malware code.

Aim:

To perform static and basic dynamic analysis of a benign sample binary using Linux command-line tools (file, strings, readelf, objdump, strace, ltrace, lsof) and document observable behaviors such as file I/O, dynamic symbol resolution, and network activity (if any).

PROCEDURE:

The project provides a hands-on guide for malware analysis using Ghidra, teaching users to dissect binaries, understand assembly and high-level behaviors, and identify malicious functionalities. It includes step-by-step tutorials, automation scripts, safe sample binaries, and reporting templates to practice analysis in a controlled environment.



```
(kali㉿kali)-[~/ghidra_cli_practice/samples]
mkdir -p ~/ghidra_cli_practice/output/report_evidence
../output/* ~/ghidra_cli_practice/output/report_evidence/ 2>/dev/null || true
opy created artifact too
report.txt ~/ghidra_cli_practice/output/report_evidence/ 2>/dev/null || true
-lah ~/ghidra_cli_practice/output/report_evidence

al 48K
xrwxr-x 2 kali kali 4.0K Oct 23 09:25 .
xrwxr-x 3 kali kali 4.0K Oct 23 09:25 ..
-rw-r-- 1 kali kali  337 Oct 23 09:25 ltrace_bs1.txt
-rw-r-- 1 kali kali 7.4K Oct 23 09:25 objdump_full_bs1.txt
-rw-r-- 1 kali kali 1.6K Oct 23 09:25 objdump_main_bs1.txt
-rw-r-- 1 kali kali  979 Oct 23 09:25 readelf_header_bs1.txt
-rw-r-- 1 kali kali 3.8K Oct 23 09:25 readelf_symbols_bs1.txt
-rw-r-- 1 kali kali   40 Oct 23 09:25 report.txt
-rw-r-- 1 kali kali   81 Oct 23 09:25 sha256_bs1.txt
-rw-r-- 1 kali kali 2.8K Oct 23 09:25 strace_bs1.txt
-rw-r-- 1 kali kali 1.2K Oct 23 09:25 strings_bs1.txt

(kali㉿kali)-[~/ghidra_cli_practice/samples]


(kali㉿kali)-[~/ghidra_cli_practice/samples]
cd ~/ghidra_cli_practice/output/report_evidence


(kali㉿kali)-[~/ghidra_cli_practice/output/report_evidence]
cat > analysis_report.md << 'EOF'
edoc> # ✅ Aim
analyze a benign sample binary using Linux commands (`file`, `strings`, `strace`, `ltrace`, `ps`, `lsof`) and o
rve its behavior.

📝 Procedure
Created two sample C programs: `benign_sample1.c` and `benign_sample2.c`.
Compiled the programs:
```bash
gcc -o benign_sample1 benign_sample1.c
gcc -o benign_sample2 benign_sample2.c -ldl
edoc>
edoc> EOF
```

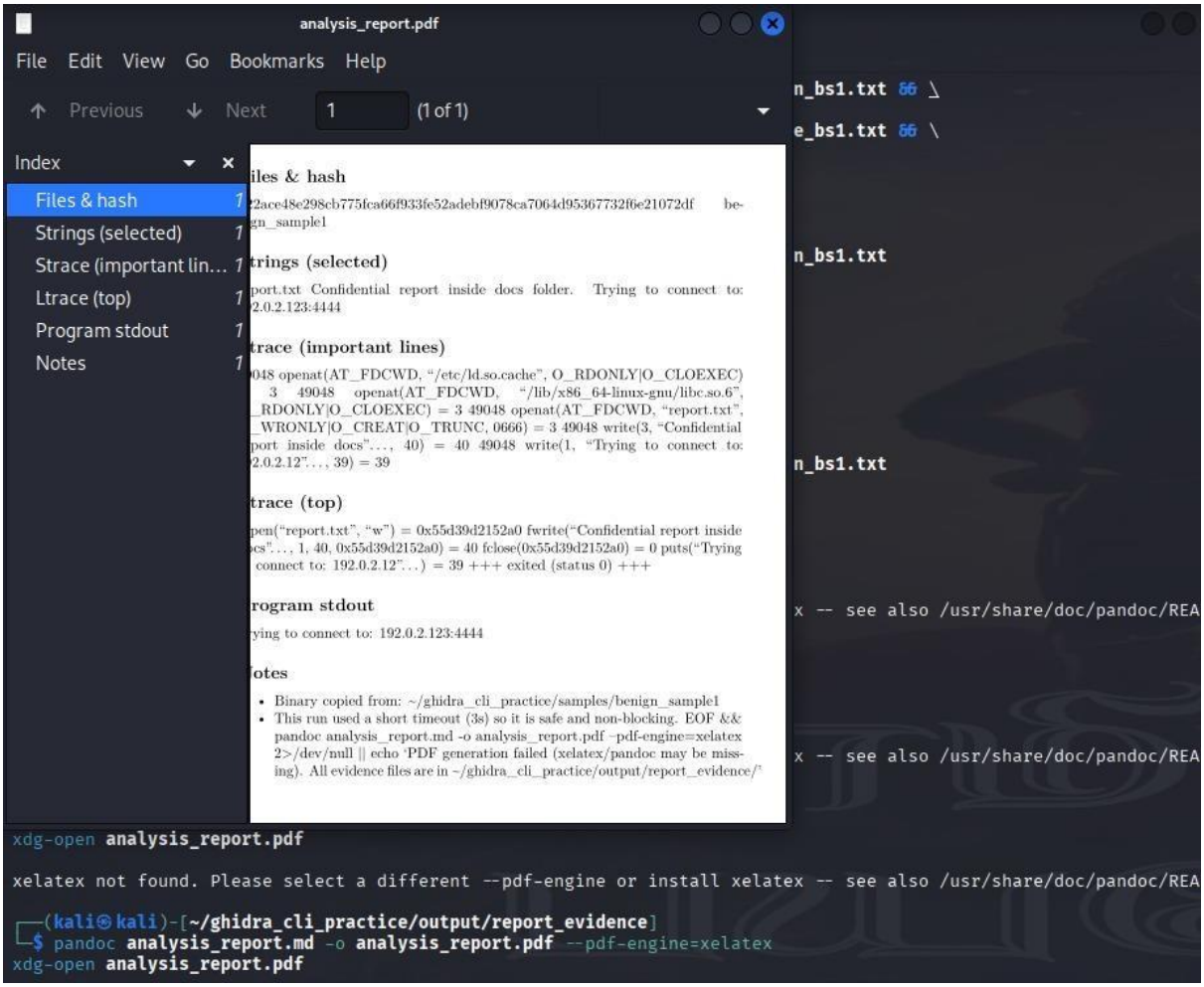
```
┌──(kali㉿kali)-[~/ghidra_cli_practice/output/report_evidence]
└─$ # Go to report folder
cd ~/ghidra_cli_practice/output/report_evidence && \

# Run the binary in background and capture its PID
./benign_sample1 & PID=$! && \

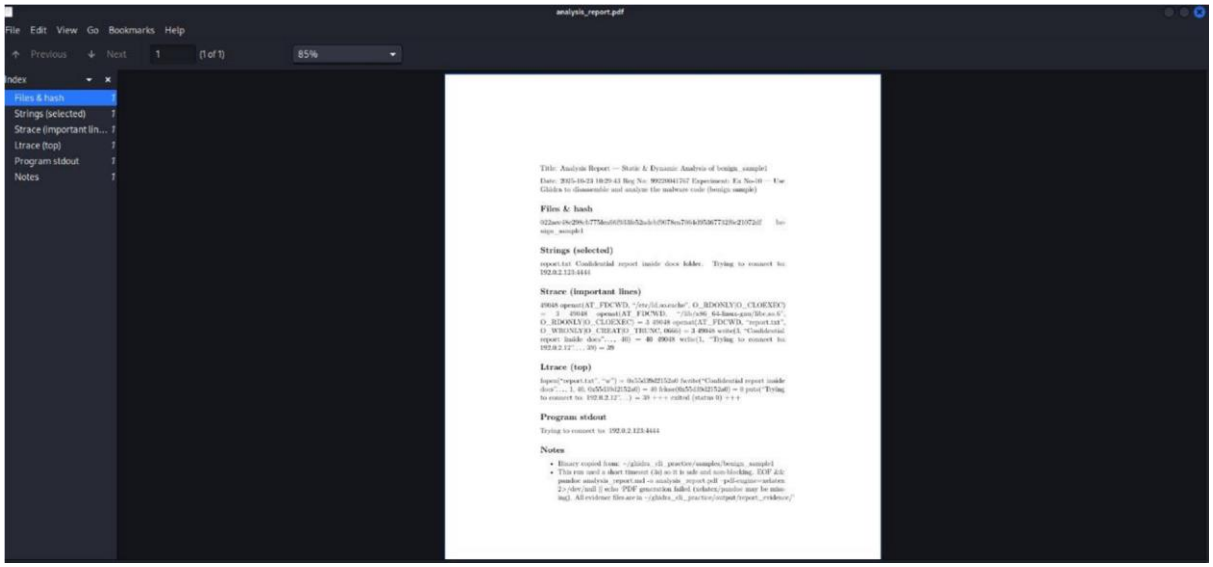
# Collect dynamic traces
strace -o strace_bs1.txt -f -p $PID & wait $PID && \
ltrace -o ltrace_bs1.txt -f -p $PID & wait $PID && \
```

```
# Collect static analysis
file benign_sample1 > file_bs1.txt && \
sha256sum benign_sample1 > sha256_bs1.txt && \
strings -n 4 benign_sample1 > strings_bs1.txt && \
readelf -h benign_sample1 > readelf_header_bs1.txt && \
readelf -s benign_sample1 > readelf_symbols_bs1.txt && \
objdump -d -M intel benign_sample1 > objdump_full_bs1.txt && \
objdump -d -M intel benign_sample1 | sed -n '/<main>:/,/^$/p' > objdump_main_bs1.txt && \
```
```

```
┌──(kali㉿kali)-[~/ghidra_cli_practice/output/report_evidence]
└─$ sudo kill -9 $PID 2>/dev/null
```



RESULT :




```
┌──(kali㉿kali)-[~]
└─$ cd ~/ghidra_cli_practice/output/report_evidence
pwd
ls -lah

/home/kali/ghidra_cli_practice/output/report_evidence
total 132K
drwxrwxr-x 2 kali kali 4.0K Oct 23 10:31 .
drwxrwxr-x 3 kali kali 4.0K Oct 23 09:25 ..
-rw-rw-r-- 1 kali kali 1.5K Oct 23 10:29 analysis_report.md
-rw-rw-r-- 1 kali kali  64K Oct 23 10:31 analysis_report.pdf
-rwxrwxr-x 1 kali kali  16K Oct 23 10:29 benign_sample1
-rw-rw-r-- 1 kali kali  224 Oct 23 10:41 file_bs1.txt
-rw-rw-r-- 1 kali kali    0 Oct 23 10:39 ltrace_bs1.txt
-rw-rw-r-- 1 kali kali 7.4K Oct 23 10:41 objdump_full_bs1.txt
-rw-rw-r-- 1 kali kali 1.6K Oct 23 10:41 objdump_main_bs1.txt
-rw-rw-r-- 1 kali kali   39 Oct 23 10:29 prog_stdout.txt
-rw-rw-r-- 1 kali kali  979 Oct 23 10:41 readelf_header_bs1.txt
-rw-rw-r-- 1 kali kali 3.8K Oct 23 10:41 readelf_symbols_bs1.txt
-rw-rw-r-- 1 kali kali   40 Oct 23 10:38 report.txt
-rw-rw-r-- 1 kali kali   81 Oct 23 10:41 sha256_bs1.txt
-rw-rw-r-- 1 kali kali    0 Oct 23 10:39 strace_bs1.txt
-rw-rw-r-- 1 kali kali 1.2K Oct 23 10:41 strings_bs1.txt

┌──(kali㉿kali)-[~/ghidra_cli_practice/output/report_evidence]
└─$ cat >> analysis_report.md <<'EOF'

## Appendix (evidence files attached)
- report.txt
- strings_bs1.txt
- sha256_bs1.txt
- strace_bs1.txt
- ltrace_bs1.txt
- objdump_main_bs1.txt
- readelf_header_bs1.txt

EOF


┌──(kali㉿kali)-[~/ghidra_cli_practice/output/report_evidence]
└─$ tail -n 40 analysis_report.md


## Files & hash
022ace48e298cb775fca66f933fe52adebf9078ca7064d95367732f6e21072df  benign_sample1

## Strings (selected)
report.txt
Confidential report inside docs folder.
Trying to connect to: 192.0.2.123:4444
```


┌──(kali㉿kali)-[~/ghidra_cli_practice/output/report_evidence]
└─$ tail -n 40 analysis_report.md
Files & hash
022ace48e298cb775fca66f933fe52adebf9078ca7064d95367732f6e21072df benign_sample1
Strings (selected)
report.txt
Confidential report inside docs folder.
Trying to connect to: 192.0.2.123:4444
Strace (important lines)
49048 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
49048 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
49048 openat(AT_FDCWD, "report.txt", O_WRONLY|O_CREAT|O_TRUNC, 0666) = 3
49048 write(3, "Confidential report inside docs "..., 40) = 40
49048 write(1, "Trying to connect to: 192.0.2.12"..., 39) = 39
Ltrace (top)
fopen("report.txt", "w")   = 0×55d39d2152a0
fwrite("Confidential report inside docs "..., 1, 40, 0×55d39d2152a0) = 40
fclose(0×55d39d2152a0)   = 0
puts("Trying to connect to: 192.0.2.12"...)   = 39
+++ exited (status 0) +++
Program stdout
Trying to connect to: 192.0.2.123:4444
Notes
- Binary copied from: ~/ghidra_cli_practice/samples/benign_sample1
- This run used a short timeout (3s) so it is safe and non-blocking.
EOF && pandoc analysis_report.md -o analysis_report.pdf --pdf-engine=xelatex 2>/dev/null || echo 'PDF generation f
ailed (xelatex/pandoc may be missing). All evidence files are in ~/ghidra_cli_practice/output/report_evidence/'
Appendix (evidence files attached)
- report.txt
- strings_bs1.txt
- sha256_bs1.txt
- strace_bs1.txt
- ltrace_bs1.txt
- objdump_main_bs1.txt
- readelf_header_bs1.txt