

# System Security Lab Report

Submitted By: Shudarsan Regmi  
CH.SC.U4CYS23055

## Lab Exercise 1

### Implementing the discretionary access control (DAC) mechanism in operating Systems (Linux)

#### Checking for a file Permission

```
ls -lah
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.064s)
ls -lah file.txt
-rw-rw-r-- 1 aparichit aparichit 0 Dec 9 13:36 file.txt
```

#### \*\*No. of hard links \*\*

```
file.txt
no. of hard links
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.064s)
ls -lah
total 8.0K
drwxrwxr-x 2 aparichit aparichit 4.0K Dec 9 13:36 .
drwxrwxr-x 3 aparichit aparichit 4.0K Dec 9 13:35 ..
-rw-rw-r-- 1 aparichit aparichit 0 Dec 9 13:36 file.txt
A hard link is
essentially a reference
or pointer to the same
inode (underlying file
data).
```

Directories typically have multiple links because of their inherent structure (. and ..), whereas regular files usually have one unless explicitly linked using the ln command.

#### Checking for parent directory permission

```
ls -ld
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.063s)
ls -ld
drwxrwxr-x 2 aparichit aparichit 4096 Dec 9 13:36 .
```

## Changing the permission of a file

- u -> Users
- o -> Owner
- g -> Group
- a -> All

### Changing file permissions

```
chmod u+x file.txt
```

```
chmod g+x file2.txt
```

```
chmod o+x file2.txt  
chmod o-x file2.txt  
chmod o+rwx file.txt  
chmod o-rwx file.txt
```

```
chmod a-rwx file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.064s )
ls -lah
total 8.0K
drwxrwxr-x 2 aparichit aparichit 4.0K Dec  9 13:36 .
drwxrwxr-x 3 aparichit aparichit 4.0K Dec  9 13:35 ..
-rw-rw-r-- 1 aparichit aparichit    0 Dec  9 13:36 file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.061s )
chmod o+x file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.064s )
chmod g+x file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.065s )
ls -lah
total 8.0K
drwxrwxr-x 2 aparichit aparichit 4.0K Dec  9 13:36 .
drwxrwxr-x 3 aparichit aparichit 4.0K Dec  9 13:35 ..
-rwxrwxr-x 1 aparichit aparichit    0 Dec  9 13:36 file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.061s )
chmod g-x file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.063s )
ls -lah
total 8.0K
drwxrwxr-x 2 aparichit aparichit 4.0K Dec  9 13:36 .
drwxrwxr-x 3 aparichit aparichit 4.0K Dec  9 13:35 ..
----- 1 aparichit aparichit    0 Dec  9 13:36 file.txt
```

## Using chmod number with permission

```
chmod uog file.txt
u = rwx bits (111) = 7
o = rwx bits (111) = 7
g = rwx bits (111) = 7
chmod 777
```

## Changing permission recursively

```
chmod -R uog file.txt
chmod -R 444 file.txt
```

## Changing the Ownership of a file

### Changing the user

```
sudo chown user file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.063s)
ls
file.txt

~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.055s)
chown user1 file.txt
chown: changing ownership of 'file.txt': Operation not permitted

② Did you mean: sudo chown user1 file.txt

~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.073s)
sudo !!
sudo chown user1 file.txt

~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground (0.062s)
ls -lah
total 8.0K
drwxrwxr-x 2 aparichit aparichit 4.0K Dec  9 13:36 .
drwxrwxr-x 3 aparichit aparichit 4.0K Dec  9 13:35 ..
----- 1 user1      aparichit    0 Dec  9 13:36 file.txt
```

## Changing the group

groups aparichit

```
sudo chgrp developers file.txt
```

## changing both the user and group

```
sudo chown user:developers file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.074s )
sudo chown user1:developers file.txt
```

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground ( 0.064s )
ls -lah
total 8.0K
drwxrwxr-x 2 aparichit aparichit 4.0K Dec  9 13:36 .
drwxrwxr-x 3 aparichit aparichit 4.0K Dec  9 13:35 ..
----- 1 user1      developers    0 Dec  9 13:36 file.txt
```

Toggle on / off natural language detection in the command line input.

```
~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Lab0/playground
```

## SUID, SGID and Sticky Bits Summary

### Summary of Symbols in File Permissions

Permission	Symbol in <code>ls -l</code>	Effect
SUID	<code>s</code> in owner exec ( <code>rws</code> )	File executes with owner's privileges.
SGID	<code>s</code> in group exec ( <code>r-s</code> )	File executes with group's privileges; directory ensures group consistency.
Sticky Bit	<code>t</code> in others exec ( <code>rwt</code> )	Only owner can delete/modify their files in the directory.

## Special permissions in Linux (SUID, SGID, sticky)

### Understanding SUID, SGID, and the Sticky Bit in Linux

In Linux, **file permissions** determine who can read, write, or execute a file or directory. However, there are three **special permission bits** that modify how execution or access is handled:

1. **SUID (Set User ID)**
2. **SGID (Set Group ID)**
3. **Sticky Bit**

These special bits are mainly used for **security and access control**. Let's dive into each in detail.

### 1. SUID (Set User ID)

**SUID (Set User ID) is a special permission that allows a user to execute a file with the permissions of the file's owner.**

This is useful when a normal user needs to execute a program that requires **elevated privileges**, such as accessing hardware or modifying system files.

## How It Works

- Normally, when you execute a file, it runs with **your** privileges.
- If the **SUID bit** is set, the file runs with the **owner's** privileges instead of the user's.
- The owner is usually **root** for system binaries.

## Example Use Case

A classic example is the `/usr/bin/passwd` command, which allows users to change their passwords.

```
ls -l /usr/bin/passwd
```

Output:

```
-rwsr-xr-x 1 root root 54256 Feb 5 2024 /usr/bin/passwd
```

```
base ~/rough/syseclab/finalprac/static (0.124s)
ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 59976 Feb 6 2024 /usr/bin/passwd

base ~/rough/syseclab/finalprac/static
```

- The **s** in `rwsr-xr-x` means the SUID bit is set.
- The file is owned by **root**, but any user can execute it with **root privileges**.
- This allows non-root users to change their passwords, even though password files are system-protected.

## Setting the SUID Bit

You can set the **SUID** bit using the `chmod` command.

```
chmod u+s filename
```

Example:

```
chmod u+s myscript.sh
```

To remove it:

```
chmod u-s myscript.sh
```

```
base ~/rough/syseclab/finalprac/suid (0.124s)
ls -l
total 4
-rw-rw-r-- 1 aparichit aparichit 26 Mar 3 11:22 myscript.sh
```

```
base ~/rough/syseclab/finalprac/suid (2.238s) ⌘ ⌘ ⌘ ⌘ ⌘
sudo chmod u+s myscript.sh
[sudo] password for aparichit:
```

```
base ~/rough/syseclab/finalprac/suid (0.13s)
ls -l
total 4
-rwSr-wr-- 1 aparichit aparichit 26 Mar 3 11:22 myscript.sh
```

```
base ~/rough/syseclab/finalprac/suid
```

## Security Concerns

- SUID can be a **security risk** if applied to insecure binaries.
- Attackers can exploit vulnerable SUID binaries to escalate privileges.

To find all SUID files on your system:

```
find / -perm -4000 -type f 2>/dev/null
```

```
base ~/rough/syseclab/finalprac/suid (56.421s)
find / -perm -4000 -type f 2>/dev/null
/home/aparichit/rough/syseclab/finalprac/suid/myscript.sh
/snap/brave/482/opt/brave.com/brave/chrome-sandbox
/snap/brave/480/opt/brave.com/brave/chrome-sandbox
/snap/core20/2434/usr/bin/chfn
/snap/core20/2434/usr/bin/chsh
/snap/core20/2434/usr/bin/gpasswd
/snap/core20/2434/usr/bin/mount
/snap/core20/2434/usr/bin/newgrp
/snap/core20/2434/usr/bin/passwd
/snap/core20/2434/usr/bin/su
/snap/core20/2434/usr/bin/sudo
/snap/core20/2434/usr/bin/umount
/snap/core20/2434/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core20/2434/usr/lib/openssh/ssh-keysign
/snap/core20/2496/usr/bin/chfn
/snap/core20/2496/usr/bin/chsh
/snap/core20/2496/usr/bin/gpasswd
/snap/core20/2496/usr/bin/mount
/snap/core20/2496/usr/bin/newgrp

base ~/rough/syseclab/finalprac/suid
```

## 2. SGID (Set Group ID)

**SGID (Set Group ID) works similarly to SUID but for groups.**

- When applied to **files**, it allows execution with the **group's** permissions.
- When applied to **directories**, it ensures that all newly created files **inherit the directory's group**.

### SGID on Files

Example: `/usr/bin/locate`

```
ls -l /usr/bin/locate
```

Output:

```
-rwx--S--x 1 root locate 43656 Feb 5 2024 /usr/bin/locate
```

- The **S** in **--S--X** indicates SGID is set.
- The file is owned by the **root** user but belongs to the **locate** group.
- Any user executing this file will run it with the **locate group's** permissions.

### Setting SGID on Files

```
chmod g+s filename
```

Example:

```
chmod g+s myscript.sh
```

The terminal window shows the command `chmod g+s myscript.sh` being run, followed by an `ls -lah` command. The output shows a file named `myscript.sh` with permissions `-rwSrSr--`. A red arrow points from the word "suid" in the prompt to the first 's' in the permission string. Another red arrow points from the word "sgid" in the prompt to the second 's' in the permission string.

```
myscript.sh
base ~/rough/syseclab/finalprac/suid (0.118s)
chmod g+s myscript.sh

base ~/rough/syseclab/finalprac/suid (0.12s)
ls -lah
total 12K
drwxrwxr-x 2 aparichit aparichit 4.0K Mar 3 11:22 .
drwxrwxr-x 4 aparichit aparichit 4.0K Mar 3 11:22 ..
-rwSrSr-- 1 aparichit aparichit 26 Mar 3 11:22 myscript.sh

base ~/rough/syseclab/finalprac/suid
lssuid sgid
```

To remove it:

```
chmod g-s myscript.sh
```

The terminal window shows the command `chmod g-s myscript.sh` being run, followed by an `ls -l` command. The output shows a file named `myscript.sh` with permissions `-rwSr-wr--`.

```
base ~/rough/syseclab/finalprac/suid (0.119s)
chmod g-s myscript.sh

base ~/rough/syseclab/finalprac/suid (0.131s)
ls -l
total 4
-rwSr-wr-- 1 aparichit aparichit 26 Mar 3 11:22 myscript.sh

base ~/rough/syseclab/finalprac/suid
```

## SGID on Directories

When SGID is set on a directory, **all new files created inside will inherit the directory's group**, rather than the creator's default group.

## Example Use Case

A shared group directory where multiple users collaborate.

```
mkdir /shared  
chmod 2775 /shared  
chown :developers /shared
```

- The **2** in **2775** sets the **SGID bit**.
- Now, any file created in **/shared** will **inherit the group 'developers'**, ensuring consistent access.

## Finding SGID Files

```
find / -perm -2000 -type f 2>/dev/null
```

## 3. Sticky Bit

The **Sticky Bit** is used **only on directories**.

It ensures that **only the owner of a file (or root) can delete or modify it**, even if the directory is world-writable.

### How It Works

- Normally, in a world-writable directory (**/tmp**), **any user** can delete any file.
- If the **Sticky Bit** is set, only the file **owner** (or root) can delete it.

### Example: **/tmp** Directory

```
ls -ld /tmp
```

Output:

```
drwxrwxrwt 10 root root 4096 Feb 5 2024 /tmp
```

- The **t** in **rwxrwxrwt** indicates the **Sticky Bit is set**.
- This prevents users from deleting files that **they don't own**.

### Setting the Sticky Bit

```
chmod +t directory_name
```

Example:

```
chmod +t /public
```

To remove it:

```
chmod -t /public
```

```
base ~/rough/syseclab/finalprac/suid (0.188s)
sudo chmod +t stickydir/

base ~/rough/syseclab/finalprac/suid (0.122s)
ls -l
total 8
-rwSr--r-- 1 aparichit aparichit 26 Mar 3 11:22 myscript.sh
drwxrwxr-t 2 aparichit aparichit 4096 Mar 3 11:28 stickydir

base ~/rough/syseclab/finalprac/suid
```

## Finding Directories with Sticky Bit

```
find / -perm -1000 -type d 2>/dev/null
```

## Summary Table

Bit	Symbol	Applied To	Effect
<b>SUID</b>	<b>s</b> (User)	Executable files	Runs with the <b>owner's</b> privileges instead of the user's.
<b>SGID</b>	<b>s</b> (Group)	Files & Directories	Files run with the <b>group's</b> permissions; directories inherit the group.
<b>Sticky Bit</b>	<b>t</b>	Directories	Prevents users from deleting files they don't own.

## Practical Examples

### Example 1: Creating an SUID Program

Let's create a simple **C program** that prints the `/etc/shadow` file (which stores encrypted passwords and is only readable by root).

### Step 1: Create the C File

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

int main() {
    system("cat /etc/shadow");
    return 0;
}
```

### Step 2: Compile and Set SUID

```
gcc uid_example.c -o uid_example
chmod u+s uid_example
sudo chown root uid_example
```

Now, even if a **normal user** runs `./uid_example`, it will execute as **root**.

---

### Example 2: Setting SGID on a Shared Directory

If multiple users need access to a project folder:

```
sudo mkdir /project
sudo chown :developers /project
sudo chmod 2775 /project
```

- New files inside `/project` will inherit the **developers** group.
- 

### Example 3: Secure Temporary Directory with Sticky Bit

To create a shared directory where users **cannot delete each other's files**:

```
sudo mkdir /public
sudo chmod 1777 /public
```

Now, `/public` is writable by everyone, but users **can only delete their own files**.

---

## Conclusion

- **SUID** allows **privilege escalation** for executables (e.g., `passwd`).
- **SGID** is useful for **group collaboration** (e.g., shared directories).
- **Sticky Bit** is essential for **public directories** (e.g., `/tmp`).

Each of these **special permission bits** improves **security and usability** but must be **used carefully** to avoid security risks.

## Creating Static Libraries in C - Lab 2

### What is a Static Library?

A static library is a collection of object files (.o) bundled together in a single archive (.a) that can be linked into executables. Unlike dynamic libraries (.so or .dll), static libraries are copied into the final binary, making them self-contained.

```
finance_project/
|__ include/
|   |__ finance.h
|__ src/
|   |__ finance.c
|__ lib/
|__ main.c
|__ Makefile
```

### Codes

```
EXPLORER ... C main.c X C finance.c M Makefile ...
LABSTATIC finance_project > C main.c > main()
1 #include <stdio.h>
2 #include "finance.h"
3
4
5 int main() {
6     double p = 1000.0, r = 5.0;
7     int t = 3;
8
9     printf("Simple Interest: %.2f \n", simple_interest(p, r, t));
10    printf("Compound Interest: %.2f \n", compound_interest(p, r, t));
11
12    return 0;
13 }

C finance.c X C finance.h ...
finance_project > src > C finance.c > simple_interest(double, int, double)
1 #include "finance.h"
2 #include <math.h>
3
4 double simple_interest(double principal, int time,
5 |    return principal * time * rate;
6 }
7
8 double compound_interest(double principal, int time
9 |    return principal * pow(1 + rate, time) - principal;
10 }
11

C finance.h X
finance_project > include > C finance.h > ...
1 #ifndef FINANCE_H
2 #define FINANCE_H
3
4 double simple_interest(double principal, int time,
5 double compound_interest(double principal, int time
6
7#endif

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE COMMENTS
rm -f src/*.o lib/*.a finance_app
aparichit@SUSHANKYA:~/Desktop/SemIV/Vault/SystemSecurity/Labs/LabStatic/finance_project$ make
gcc -c src/finance.c -o src/finance.o -Iinclude
ar rcs lib/libfinance.a src/finance.o
gcc main.c -o finance_app -Iinclude -Llib -lfinance -lm
aparichit@SUSHANKYA:~/Desktop/SemIV/Vault/SystemSecurity/Labs/LabStatic/finance project$ 
```

## main.c

```
#include <stdio.h>
#include "finance.h"

int main() {
    double p = 1000.0, r = 5.0;
    int t = 3;

    printf("Simple Interest: %2.f \n", simple_interest(p, t, r));
    printf("Compound Interest: %2.f \n", compound_interest(p, t, r));

    return 0;
}
```

## finance.h

```
#ifndef FINANCE_H
#define FINANCE_H

double simple_interest(double principal, int time, double rate);
double compound_interest(double principal, int time, double rate);

#endif
```

## finance.c

```
#include "finance.h"
#include <math.h>

double simple_interest(double principal, int time, double rate) {
    return principal * time * rate;
}

double compound_interest(double principal, int time, double rate) {
    return principal * pow(1 + rate, time) - principal;
```

```
}
```

```
# Compiler
CC = gcc

# Directories
SRC_DIR = src
INCLUDE_DIR = include
LIB_DIR = lib

# Source and Object Files
SRC_FILES = $(SRC_DIR)/finance.c
OBJ_FILES = $(SRC_DIR)/finance.o

# Static Library Name
LIB_NAME = libfinance.a

# Executable Name
TARGET = finance_app

# Compiler and Linker Flags
CFLAGS = -I$(INCLUDE_DIR)
LDFLAGS = -L$(LIB_DIR) -lfinance -lm

# Default target
all: $(TARGET)

# Build the Static Library
$(LIB_DIR)/$(LIB_NAME): $(OBJ_FILES)
    ar rcs $@ $^

# Compile Object Files
$(SRC_DIR)/finance.o: $(SRC_DIR)/finance.c $(INCLUDE_DIR)/finance.h
    $(CC) -c $< -o $@ $(CFLAGS)

# Build the Executable
$(TARGET): main.c $(LIB_DIR)/$(LIB_NAME)
    $(CC) main.c -o $(TARGET) $(CFLAGS) $(LDFLAGS)

# Force re-linking every time
# .PHONY: all clean

# Clean up generated files
clean:
    rm -f $(SRC_DIR)/*.o $(LIB_DIR)/*.a $(TARGET)
```

## Compiling the library

```
gcc -c src/finance.c -o src/finance.o -Iinclude
```

## Creating .a archive from object files

```
ar rcs lib/libfinance.a src/finance.o # use this  
ar rcs lib/finance.a src/finance.o # this had created problem, i.e. giving  
non-standard name to the library file
```

## Checking what object files does the archive contains

```
ar -t lib/libfinance.a
```

## Final compilation command

```
gcc main.c -Llib -l:finance.a -lm -Iinclude -o finance_app # if the library  
name is non standard  
gcc main.c -Llib -lfinance -lm -Iinclude -o finance_app # if follows  
standard naming of static file like. libfinance, libcalculator, etc.
```

Static libraries in Unix-like systems follow the lib.a convention because the linker (ld) automatically searches for files prefixed with lib when using -l, and omitting this prefix requires explicitly specifying the filename with -l:.a.

# Creating Dynamic Libraries - Lab 3

## Creating and Using a Dynamic Library in C

The process of creating and utilizing a **dynamic library (shared library)** in C involves several steps. The development takes place in a **Linux** environment, where the shared object file (**.so**) is compiled and linked with a program.

### 1. Development of the Library

A dynamic library typically consists of reusable functions that can be shared across multiple programs. In this case, a **mathematical library** was developed.

```

# ifndef MYMATH_H
# define MYMATH_H

// Function prototypes
int add(int a, int b);
int subtract(int a, int b);
int multiply(int a, int b);
double divide(int a, int b);

#endif
mymath.h

mymath.h
#include <stdio.h>

// Function to add two numbers
int add(int a, int b) {
    return a + b;
}

// Function to subtract two numbers
int subtract(int a, int b) {
    return a - b;
}

// Function to multiply two numbers
int multiply(int a, int b) {
    return a * b;
}
mymath.c
:split mymath.h

```

11.0-1      All

```

#include <stdio.h>
#include "mymath.h" // Include the header file

int main() {
    int a = 10, b = 5;

    printf("Addition: %d\n", add(a, b));
    printf("Subtraction: %d\n", subtract(a, b));
    printf("Multiplication: %d\n", multiply(a, b));
    printf("Division: %.2f\n", divide(a, b));

    return 0;
}
main.c


```

1,1      Top      1,1      All

## 1.1. Implementation of the Library ([mymath.c](#))

A source file named [mymath.c](#) was created to define mathematical functions. The functions included in the library perform basic arithmetic operations such as addition, subtraction, multiplication, and division. The implementation is as follows:

```

#include <stdio.h>

int add(int a, int b) {
    return a + b;
}

int subtract(int a, int b) {
    return a - b;
}

int multiply(int a, int b) {
    return a * b;
}

double divide(int a, int b) {
    if (b == 0) {
        printf("Error: Division by zero!\n");
        return 0.0;
    }
    return (double)a / b;
}

```

## 1.2. Declaration of Function Prototypes ([mymath.h](#))

A header file (`mymath.h`) was created to declare the function prototypes. This header file allows other programs to access the functions contained within the library.

```
#ifndef MYMATH_H
#define MYMATH_H

int add(int a, int b);
int subtract(int a, int b);
int multiply(int a, int b);
double divide(int a, int b);

#endif
```

## 2. Compilation of the Shared Library

To generate the shared object file, the source code was compiled using **GCC**. The following commands were used:

```
gcc -fPIC -c mymath.c -o mymath.o
gcc -shared -o libmymath.so mymath.o
```

```
base ~/rough/syseclab/finalprac/dynamiclib (0.247s)
gcc -fPIC -c mymath.c -o mymath.o
```

```
base ~/rough/syseclab/finalprac/dynamiclib (0.219s) ⌂ ⌂ ⌂ ⌂
gcc -shared -o libmymath.so mymath.o
```

```
base ~/rough/syseclab/finalprac/dynamiclib (3.195s)
```

- The `-fPIC` flag was used to generate **position-independent code**, which is necessary for shared libraries.
- The `-c` flag compiled the source file into an object file without linking.
- The `-shared` flag was used to create a **shared object file (.so)**.

As a result of this process, a file named `libmymath.so` was successfully generated.

## 3. Utilization of the Shared Library

To verify the functionality of the newly created library, a **C program (main.c)** was written.

### 3.1. Implementation of the Main Program (`main.c`)

A program was created to utilize the functions from `libmymath.so`. The corresponding implementation is shown below:

```
#include <stdio.h>
#include "mymath.h"

int main() {
    int a = 10, b = 5;

    printf("Addition: %d\n", add(a, b));
    printf("Subtraction: %d\n", subtract(a, b));
    printf("Multiplication: %d\n", multiply(a, b));
    printf("Division: %.2f\n", divide(a, b));

    return 0;
}
```

### 3.2. Compilation of the Main Program

The main program was compiled with the following command:

```
gcc -o main main.c -L. -lmymath
```

```
base ~/rough/syseclab/finalprac/dynamiclib (0.267s)
gcc -o main main.c -L. -lmymath

base ~/rough/syseclab/finalprac/dynamiclib (0.122s)
ls
libmymath.so  main  main.c  mymath.c  mymath.h  mymath.o
```

- The `-L.` option specified the current directory as the location of the library.
- The `-lmymath` option linked the program against `libmymath.so`.

## 4. Execution and Library Path Configuration

During execution, an error was encountered due to the **shared library not being found** by the system. The following message was displayed:

```
error while loading shared libraries: libmymath.so: cannot open shared
object file: No such file or directory
```

To resolve this issue, the **LD\_LIBRARY\_PATH** environment variable was modified temporarily:

```
export LD_LIBRARY_PATH=.:$LD_LIBRARY_PATH  
./main
```

The screenshot shows a terminal window with the following content:

```
base ~/rough/syseclab/finalprac/dynamiclib (0.125s) ✖ ↴ ⌂ ⌂ ⌂  
. ./main  
. ./main: error while loading shared libraries: libmymath.so: cannot open shared object file: No such file or directory  
② Did you mean: man  
  
base ~/rough/syseclab/finalprac/dynamiclib (0.12s)  
export LD_LIBRARY_PATH=.:$LD_LIBRARY_PATH  
  
base ~/rough/syseclab/finalprac/dynamiclib (0.134s)  
. ./main  
Addition: 15  
Subtraction: 5  
Multiplication: 50  
Division: 2.00  
  
base ~/rough/syseclab/finalprac/dynamiclib
```

Alternatively, to make the library accessible system-wide, it was moved to [/usr/local/lib](#) and registered with the system:

```
sudo cp libmymath.so /usr/local/lib  
sudo ldconfig
```

Following this step, the program was executed without requiring additional environment variables.

## 5. Output Verification

After successful execution, the following output was produced:

```
Addition: 15  
Subtraction: 5  
Multiplication: 50  
Division: 2.00
```

```
base ~/rough/syseclab/finalprac/dynamiclib (0.406s)
sudo cp libmymath.so /usr/local/lib
sudo ldconfig
```

```
base ~/rough/syseclab/finalprac/dynamiclib (0.094s)
./main
Addition: 15
Subtraction: 5
Multiplication: 50
Division: 2.00
```

```
base ~/rough/syseclab/finalprac/dynamiclib
```

The expected results were observed, confirming the successful creation and integration of the **dynamic library** in C.

## Conclusion

A **shared library** (.so file) was successfully developed, compiled, and integrated into a program. The necessary steps included defining functions, generating the library, linking it with a program, and ensuring proper execution by configuring the shared library path. The approach demonstrated the modular nature of dynamic libraries, allowing multiple programs to use the same functions without requiring static linking.

# To make use of make file for compilation - Lab 4

## Theory

When working on a C project with multiple source files, manually compiling each file can be inefficient. The **make** tool helps automate this process by only recompiling files that have changed, using timestamps.

### How Does **make** Track Changes?

- **make** checks **timestamps** of files.
- If a .c file has been **modified after** its corresponding .o file was last compiled, **make** recompiles only that file.
- This helps in **incremental compilation**, making the build process faster.

## Project Directory Structure

```
project/
|--- Makefile
```

```
└── main.c
└── calculator.c
└── calculator.h
```

The screenshot shows a code editor interface with four tabs open:

- Makefile**: Contains the build configuration for the project.
- main.c**: Contains the main function logic.
- calculator.c**: Contains two functions: add and subtract.
- calculator.h**: Contains the function prototypes for add and subtract.

The Makefile includes compiler settings (CC = gcc), compiler flags (CFLAGS = -g -Wall -Wextra -std=c11), target executable (TARGET = calculator), source files (SRC = main.c calculator.c), object files (OBJ = \$(SRC:.c=.o)), and a default rule for building the executable.

## Makefile

```
# Compiler
CC = gcc

# Compiler flags
CFLAGS = -g -Wall -Wextra -std=c11

# Target executable
TARGET = calculator

# Source and object files
SRC = main.c calculator.c
OBJ = $(SRC:.c=.o)

# Default rule (build the executable)
$(TARGET) : $(OBJ)
    $(CC) $(CFLAGS) -o $(TARGET) $(OBJ)

# Rule to compile C files into object files
%.o : %.c
    $(CC) $(CFLAGS) -c $< -o $@

# Clean rule to remove generated files
clean:
    rm -f $(TARGET) $(OBJ)
```

## Source Code

main.c

```
#include <stdio.h>
#include "calculator.h"

int main() {
    int a = 5, b = 3;
    printf("Sum: %d\n", add(a, b));
    printf("Difference: %d\n", subtract(a, b));
    return 0;
}
```

calculator.c

```
#include "calculator.h"

int add(int a, int b) {
    return a + b;
}

int subtract(int a, int b) {
    return a - b;
}
```

calculator.h

```
#ifndef CALCULATOR_H
#define CALCULATOR_H

int add(int a, int b);
int subtract(int a, int b);

#endif // CALCULATOR_H
```

```
aparichit@SUSHANKYA:~/rough/syseclab/makefile$ make
gcc -g -Wall -Wextra -std=c11 -c main.c -o main.o
gcc -g -Wall -Wextra -std=c11 -c calculator.c -o calculator.o making
gcc -g -Wall -Wextra -std=c11 -o calculator main.o calculator.o
aparichit@SUSHANKYA:~/rough/syseclab/makefile$ ls      after compilation
calculator calculator.c calculator.h calculator.o main.c main.o Makefile
aparichit@SUSHANKYA:~/rough/syseclab/makefile$ ./calculator
Sum: 8                                         running
Difference: 2
aparichit@SUSHANKYA:~/rough/syseclab/makefile$ make clean
rm -f calculator main.o calculator.o           cleaning
aparichit@SUSHANKYA:~/rough/syseclab/makefile$ ls
calculator.c calculator.h main.c Makefile      after cleanup
aparichit@SUSHANKYA:~/rough/syseclab/makefile$ 
```

## Building the Project

1. Open a terminal and navigate to the project directory:

```
cd project
```

2. Run `make`:

```
make
```

- This will compile `main.c` and `calculator.c`, producing the `calculator` executable.

3. Execute the program:

```
./calculator
```

- Expected output:

```
Sum: 8
Difference: 2
```

4. To clean up the compiled files:

```
make clean
```

# To create isolated environment in Linux using chroot - Lab 5

Create the directory setup

```
mkdir -p newroot/{bin,lib,lib64}
```

Copy the required binaries

```
cp -v /bin/{bash,ls,cp} /newroot/bin
```

Find the dynamic dependencies for each of above binaries

```
ldd /bin/bash  
ldd /bin/ls  
ldd /bin/pwd
```

## Output

```
ldd /bin/bash  
    linux-vdso.so.1 (0x00007ffd8a7d8000) # no need to add this  
    libtinfo.so.6 => /lib/x86_64-linux-gnu/libtinfo.so.6  
    (0x00007ec9dcddc000)  
    libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007ec9dca00000)  
    /lib64/ld-linux-x86-64.so.2 (0x00007ec9dcf8c000)  
  
ldd /bin/ls  
    linux-vdso.so.1 (0x00007ffe95355000)  
    libselinux.so.1 => /lib/x86_64-linux-gnu/libselinux.so.1  
    (0x0000753a02ebd000)  
    libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x0000753a02c00000)  
    libpcre2-8.so.0 => /lib/x86_64-linux-gnu/libpcre2-8.so.0  
    (0x0000753a02b69000)  
    /lib64/ld-linux-x86-64.so.2 (0x0000753a02f2a000)
```

```
~/rough/syseclab (0.214s)
ldd /bin/ls
  linux-vdso.so.1 (0x00007ffe95355000)
  libselinux.so.1 => /lib/x86_64-linux-gnu/libselinux.so.1 (0x0000753a02ebd000)
  libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x0000753a02c00000)
  libpcre2-8.so.0 => /lib/x86_64-linux-gnu/libpcre2-8.so.0 (0x0000753a02b69000)
  /lib64/ld-linux-x86-64.so.2 (0x0000753a02f2a000)
```

```
~/rough/syseclab (0.172s)
ldd /bin/bash
  linux-vdso.so.1 (0x00007ffdef05d000)
  libtinfo.so.6 => /lib/x86_64-linux-gnu/libtinfo.so.6 (0x000076c37023e000)
  libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x000076c370000000)
  /lib64/ld-linux-x86-64.so.2 (0x000076c3703ee000)
```

```
~/rough/syseclab
```

Copy all the respective files with proper directory structures

```
cp --parents /lib/x86_64-linux-gnu/libselinux.so.1 newroot/
cp --parents /lib/x86_64-linux-gnu/libc.so.6 newroot/
# copy all the dependencies like this
```

```
~/rough/syseclab (0.153s)
cp --parents /lib/x86_64-linux-gnu/libc.so.6 newroot/
```

```
~/rough/syseclab (0.144s)
cp --parents /lib/x86_64-linux-gnu/libunistring.so.2 newroot/
```

```
~/rough/syseclab (0.137s)
cp --parents /lib64/ld-linux-x86-64.so.2 newroot/
```

Finally run the chroot command

```
sudo chroot newroot/ /bin/bash
```

**Output** Gives the bash shell spawned

```
~/rough/syseclab (0.201s)
sudo chroot newroot /bin/ls
bin lib lib64

~/rough/syseclab
sudo chroot newroot /bin/bash
bash-5.1# ls
bin lib lib64
bash-5.1# █
```

## Apparmor - Lab 6

---

### Apparmor

---

Creating bash script

Choosing I

```
[(S)can system log for AppArmor events] / (F)inish
Reading log entries from /var/log/syslog.

Profiling: /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh

Please start the application to be profiled in
another window and exercise its functionality now.

Once completed, select the "Scan" option below in
order to scan the system logs for AppArmor events.

For each AppArmor event, you will be given the
opportunity to choose whether the access should be
allowed or denied.

[(S)can system log for AppArmor events] / (F)inish
Reading log entries from /var/log/syslog.

Profile: /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh
Execute: /usr/bin/touch
Severity: 3

(I)nherit / (C)hild / (N)amed / (X) ix on / (D)eny / Abo(r)t / (F)inish
█
```

**choosing I**

Creating and saving profiles

```
[1 - owner /home/*/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data1/sample.txt w,]
[2 - owner /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data1/sample.txt w,
(A)llow / [(D)eny] / (I)gnore / (Glob / Glob with (E)xtension / (N)ew / Audit) / (O)wner permissions off / Abo(r)t / (F)inish
Adding owner /home/*/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data1/sample.txt w, to profile.

= Changed Local Profiles =

The following local profiles were changed. Would you like to save them?

[1 - /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh]
(S)ave Changes / Save Selec(t)ed Profile / [(V)iew Changes] / View Changes b/w (C)lean profiles / Abo(r)t
Writing updated profile for /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh.

Profiling: /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh

Please start the application to be profiled in
another window and exercise its functionality now.

Once completed, select the "Scan" option below in
order to scan the system logs for AppArmor events.

For each AppArmor event, you will be given the
opportunity to choose whether the access should be
allowed or denied.

[(S)can system log for AppArmor events] / (F)inish
Setting /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh to enforce mode.

Reloaded AppArmor profiles in enforce mode.

Please consider contributing your new profile!
See the following wiki page for more information:
https://gitlab.com/apparmor/apparmor/wikis/Profiles

Finished generating profile for /home/aparichit/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor/data/example.sh.
aparichit@SUSHANKYA:~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ 
```

## Checking for restriction

deletion was denied, because  
this line was chosen to be denied  
during profile creation

```
data, data1,
aparichit@SUSHANKYA:~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ ./data/example.sh
This is an apparmor example
File created
./data/example.sh: line 5: /usr/bin/rm: Permission denied
File deleted
aparichit@SUSHANKYA:~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ 
```

## Changing the path of the script

```
root@SUSHANKYA: /etc/apparmor.d          ×      aparichit@SUSHANKYA: ~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ 

#!/bin/bash
echo "This is an apparmor example"
touch ./data3/sample.txt
echo "File created"
rm ./data3/sample.txt
echo "File deleted"

~
```

Entries from /var/log/syslog.

It was blocked

```
aparichit@SUSHANKYA:~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ nvim ./data/example.sh
aparichit@SUSHANKYA:~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ ./data/example.sh
This is an apparmor example
touch: cannot touch './data3/ample.txt': Permission denied
File created
./data/example.sh: line 5: /usr/bin/rm: Permission denied this was already deined
File deleted
aparichit@SUSHANKYA:~/Desktop/SemIV/SemIV_Vault/SystemSecurity/Labs/Apparmor$ 
```

## Conclusion:

Apparmor can be used to impose mandatory access control to an application or program.

## Discretionary Access Control in Database : MySQL - Lab 7

```
mysql> GRANT 'role1' to 'ma'@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> grant select on *.* to role1;
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> show grants;
+-----+
| Grants for root@localhost
+-----+
```

Login as root user

```
sudo mysql
```

Showing the current user

```
SELECT CURRENT_USER();
```

Creating user

```
CREATE USER 'ma'@'localhost' identified by 'ma@localhost';
```

```
1 row in set (0.00 sec)
```

```
mysql> CREATE USER 'ma'@'localhost' identified by 'ma@localhost';
Query OK, 0 rows affected (0.02 sec)
```

## Showing the grants for the current user

```
CREATE USER 'ma'@'localhost' identified by 'ma@localhost';
```

```
mysql> show grants for 'ma'@'localhost';
+-----+
| Grants for ma@localhost          |
+-----+
| GRANT USAGE ON *.* TO `ma`@`localhost` |
+-----+
1 row in set (0.00 sec)
```

## Creating database by newly created user

```
create database ma_ko_database
```

```
mysql> create database ma_ko_database;
ERROR 1044 (42000): Access denied for user 'ma'@'localhost' to database 'ma_ko_database'
mysql>
```

## Granting Privileges from root user

```
1 row in set (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'ma'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> show grants for 'ma'@'localhost';
+-----+
| Grants for ma@localhost |
+-----+
| GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE, CREATE ROLE, DROP ROLE ON *.* TO `ma`@`localhost` |
| GRANT APPLICATION_PASSWORD_ADMIN,AUDIT_ABORT_EXEMPT,AUDIT_ADMIN,AUTHENTICATION_POLICY_ADMIN,BACKUP_ADMIN,BINLOG_ADMIN,BINLOG_ENCRYPTION_ADMIN,CLONE_ADMIN,CONNECTION_ADMIN,ENCRYPTION_KEY_ADMIN,FIREWALL_EXEMPT,FLUSH_OPTIMIZER_COSTS,FLUSH_STATUS,FLUSH_TABLES,FLUSH_USER_RESOURCES,GROUP_REPLICATION_ADMIN,IN_GROUP_REPLICATION_STREAM,INNODB_REDO_LOG_ARCHIVE,INNODB_REDO_LOG_ENABLE,PASSWORDLESS_USER_ADMIN,PERSIST_RO_VARIABLES_ADMIN,REPLICATION_APPLIER,REPLICATION_SLAVE_ADMIN,RESOURCE_GROUP_ADMIN,RESOURCE_GROUP_USER,ROLE_ADMIN,SENSITIVE_VARIABLES_OBSERVER,SERVICE_CONNECTION_ADMIN,SESSION_VARIABLES_ADMIN,SET_USER_ID,SHOW_ROUTINE,SYSTEM_USER,SYSTEM_VARIABLES_ADMIN,TABLE_ENCRYPTION_ADMIN,TELEMETRY_LOG_ADMIN,XA_RECOVER_ADMIN ON *.* TO `ma`@`localhost` |
+-----+
```

## Creating database after the permission has been granted

```
mysql> create database ma_ko_database;
Query OK, 1 row affected (0.01 sec)
```

```
mysql> |
```

## Creating role

```
CREATE ROLE 'role1';
```

## Granting privileges to a role

```
GRANT SELECT ON *.* TO role1;
```

```
mysql> grant select on *.* to role1;
Query OK, 0 rows affected (0.01 sec)

mysql> show grants;
+-----+
| Grants for root@localhost |
+-----+
```

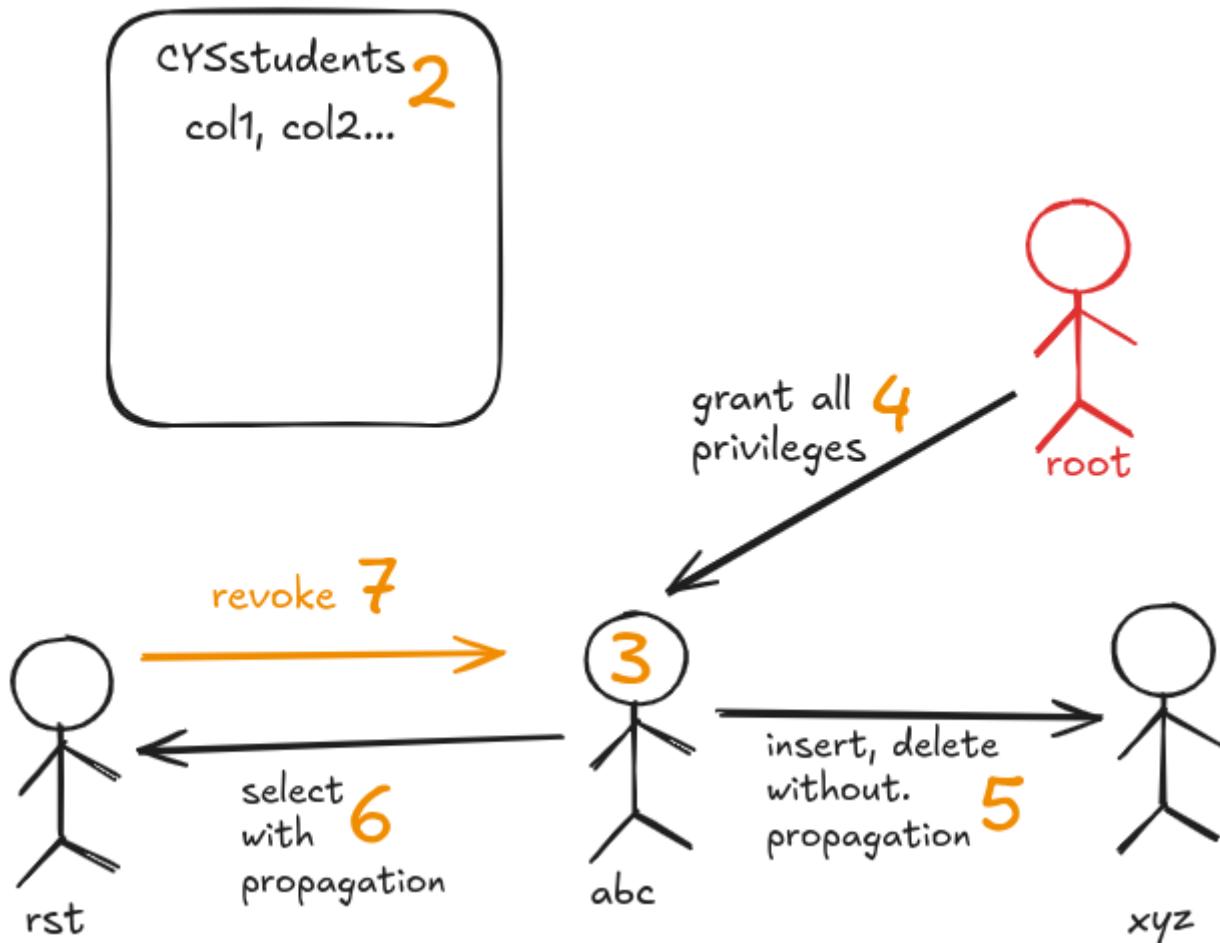
## Assigning role to a particular user

```
GRANT 'role1' TO 'ma'@'localhost';
```

## Exercise

Exercise: Create a database "AmritaChennai". Create a table "CYSSStudents" with any number of fields of your choice. Create a user account "abc" with a password. Grant all privileges to the account "abc". "abc" user account wants to grant to account "xyz" the privilege to insert and delete tuples in the table "CYSSStudents". However, "abc" does not want "xyz" to be able to propagate these privileges to additional accounts. Write the command for this and demonstrate. Suppose that "abc" wants to allow account "rst" to retrieve information from the table "CYSSStudents" and also to be able to propagate the SELECT privilege to other accounts. Write the command for this and demonstrate. Suppose that "abc" decides to revoke the SELECT privilege on the "CYSSStudents" table from "rst". Write the command for this and demonstrate.

# 1 AmritaChennai



## Step 1: Create a Database

```
CREATE DATABASE AmritaChennai;
```

## Step 2: Create a Table

```
USE AmritaChennai;

CREATE TABLE CYSStudents (
    StudentID INT PRIMARY KEY,
    StudentName VARCHAR(50),
    Course VARCHAR(30),
    Year INT
);
```

## Step 3: Create a User Account abc with a Password

```
CREATE USER 'abc'@'localhost' IDENTIFIED BY 'abc_password';
```

---

## Step 4: Grant All Privileges to User abc

```
GRANT ALL PRIVILEGES ON AmritaChennai.* TO 'abc'@'localhost';
```

---

## Step 5: Grant Insert and Delete Privileges from abc to xyz Without Grant Option

User abc logs in and executes:

```
CREATE USER 'xyz'@'localhost' IDENTIFIED BY 'xyz_password';  
GRANT INSERT, DELETE ON AmritaChennai.CYSStudents TO 'xyz'@'localhost';
```

- The absence of `WITH GRANT OPTION` ensures xyz cannot propagate these privileges.

---

## Step 6: Grant SELECT Privilege from abc to rst With Grant Option

User abc logs in and executes:

```
CREATE USER 'rst'@'localhost' IDENTIFIED BY 'rst_password';  
GRANT SELECT ON AmritaChennai.CYSStudents TO 'rst'@'localhost' WITH GRANT OPTION;
```

- `WITH GRANT OPTION` allows rst to propagate the `SELECT` privilege to other accounts.

---

## Step 7: Revoke SELECT Privilege from rst

User abc logs in and executes:

```
REVOKE GRANT OPTION FOR SELECT ON AmritaChennai.CYSStudents FROM  
'rst'@'localhost';  
REVOKE SELECT ON AmritaChennai.CYSStudents FROM 'rst'@'localhost';
```

- The first command removes the ability of rst to propagate the `SELECT` privilege.
- The second command completely revokes the `SELECT` privilege from rst.

## Roles in mysql

```
Query OK, 0 rows affected (0.01 sec)

mysql> select current_user();
+-----+
| current_user() |
+-----+
| root@localhost |
+-----+
1 row in set (0.00 sec)

mysql> create role reader;
Query OK, 0 rows affected (0.01 sec)

mysql> grant select on *.* to reader;
Query OK, 0 rows affected (0.00 sec)

mysql> show grants for reader;
+-----+
| Grants for reader@%           |
+-----+
| GRANT SELECT ON *.* TO `reader`@`%` |
+-----+
1 row in set (0.00 sec)

mysql> grant reader to 'user1'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql>
```

```
create role reader;
```

```
grant select on *.* to reader;
```

```
show grants for reader;
```

```
grant reader to 'user1'@'localhost';
```

```
revoke reader from 'user1'@'localhost';
```

# Virtual Private Database : Implementing Row Level Security in mysql - Lab 8

## Implementing Row-Level Security in MySQL



We want to store **all users' data in a single table** but **ensure that each user can only see, modify, or delete their own data**.

To achieve this, we will:

- Use a **single books table** with an **owner** column.
- Restrict access using **MySQL Views**.
- Enforce ownership using **triggers**.

### 🔧 Step 1: Clean Up (If Needed)

If you attempted this before, **delete any previous data**:

```
DROP DATABASE IF EXISTS rls_practice;
DROP USER IF EXISTS 'user1'@'localhost';
DROP USER IF EXISTS 'user2'@'localhost';
FLUSH PRIVILEGES;
```

### 📁 Step 2: Create a New Database

```
CREATE DATABASE rls_practice;
USE rls_practice;
```

#### Output

```
mysql> CREATE DATABASE rls_practice;
Query OK, 1 row affected (0.01 sec)

mysql> USE rls_practice;
Database changed
mysql> █
```

### Step 3: Create the **books** Table

This table stores all books **with an owner field** that records who inserted the book.

```
CREATE TABLE books (
    id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    author VARCHAR(255) NOT NULL,
    owner VARCHAR(50) NOT NULL
);
```

```
mysql> USE rls_practice;
Database changed
mysql> CREATE TABLE books (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     title VARCHAR(255) NOT NULL,
    ->     author VARCHAR(255) NOT NULL,
    ->     owner VARCHAR(50) NOT NULL
    -> );
Query OK, 0 rows affected (0.04 sec)
```

Output mysql>

## Step 4: Create Users

We will create **two users (user1 and user2)**.

```
CREATE USER 'user1'@'localhost' IDENTIFIED BY 'password1';
CREATE USER 'user2'@'localhost' IDENTIFIED BY 'password2';
```

Output

```
mysql> CREATE USER 'user1'@'localhost' IDENTIFIED BY 'password1';
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE USER 'user2'@'localhost' IDENTIFIED BY 'password2';
Query OK, 0 rows affected (0.04 sec)

mysql>
```

## Step 5: Enforce Ownership Using a Trigger

We create a **before-insert trigger** to automatically store the username when a book is inserted.

```
DELIMITER //
CREATE TRIGGER before_insert_books
BEFORE INSERT ON books
FOR EACH ROW
BEGIN
    SET NEW.owner = SESSION_USER(); -- Ensures the actual logged-in user is
    stored
```

```
END;
//  
DELIMITER ;
```

```
DELIMITER ;
```

## Output

```
mysql>  
mysql> DELIMITER //  
mysql> CREATE TRIGGER before_insert_books  
    -> BEFORE INSERT ON books  
    -> FOR EACH ROW  
    -> BEGIN  
    ->     SET NEW.owner = SESSION_USER(); -- Ensures the actual logged-in user is stored  
    -> END;  
    -> //  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> DELIMITER ;
```

```
DELIMITER //  
CREATE TRIGGER before_update_books  
BEFORE UPDATE ON books  
FOR EACH ROW  
BEGIN  
    IF OLD.owner != CURRENT_USER() THEN  
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Unauthorized update!';  
    END IF;  
END;  
//  
DELIMITER ;
```

## Output

```
mysql> DELIMITER //  
mysql> CREATE TRIGGER before_update_books  
    -> BEFORE UPDATE ON books  
    -> FOR EACH ROW  
    -> BEGIN  
    ->     IF OLD.owner != CURRENT_USER( ) THEN  
    ->         SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Unauthorized update!';  
    ->     END IF;  
    -> END;  
    -> //  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> DELIMITER ;  
mysql>
```

## Step 6: Create a View for Row-Level Security

Since users should **only see their own books**, we create a **view** that filters by **owner**.

```
CREATE VIEW user_books AS
SELECT id, title, author FROM books WHERE owner = SESSION_USER();
```

### Output

```
mysql>
mysql>
mysql> CREATE VIEW user_books AS
-> SELECT id, title, author FROM books WHERE owner = SESSION_USER();
Query OK, 0 rows affected (0.02 sec)

mysql>
```

## Step 7: Grant Permissions to Users

1. **Revoke direct table access** (so users cannot see everything).

```
REVOKE ALL PRIVILEGES ON books FROM 'user1'@'localhost';
REVOKE ALL PRIVILEGES ON books FROM 'user2'@'localhost';
```

2. **Grant access to the `user_books` view** (so they only see their data).

```
GRANT SELECT, INSERT, UPDATE, DELETE ON user_books TO 'user1'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON user_books TO 'user2'@'localhost';
```

### Output

```
mysql> GRANT SELECT, INSERT, UPDATE, DELETE ON user_books TO 'user1'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT SELECT, INSERT, UPDATE, DELETE ON user_books TO 'user2'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql>
```

3. **Allow inserting books** (users can insert, but ownership is handled by the trigger).

```
GRANT INSERT ON books TO 'user1'@'localhost';
GRANT INSERT ON books TO 'user2'@'localhost';
```

## Output

```
mysql> GRANT INSERT ON books TO 'user1'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT INSERT ON books TO 'user2'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

## Step 8: Testing

### Test as user1

#### Login as user1

```
mysql -u user1 -p
USE rls_practice;
```

#### Insert a book

```
INSERT INTO books (title, author) VALUES ('User1 Book', 'Author A');
```

#### Check books (should only see their own)

```
SELECT * FROM user_books;
```

#### Output should be:

```
+----+-----+-----+
| id | title      | author    |
+----+-----+-----+
| 1  | User1 Book | Author A |
+----+-----+-----+
```

### Test as user2

#### Login as user2

```
mysql -u user2 -p
USE rls_practice;
```

## Insert a book

```
INSERT INTO books (title, author) VALUES ('User2 Book', 'Author B');
```

## Check books (should only see their own)

```
SELECT * FROM user_books;
```

## Output should be:

```
+----+-----+-----+
| id | title      | author     |
+----+-----+-----+
| 2  | User2 Book | Author B |
+----+-----+-----+
```

## 🔍 Step 9: Verify Security

Login as **user1** and try to access **user2**'s data:

```
SELECT * FROM books;
```

**Access Denied!**

**Security is working!**

## Final Summary

### What We Achieved

- ✓ A **single table (books)** that stores all users' books.
- ✓ Users **can only see their own data** (enforced via a **VIEW**).
- ✓ A **trigger automatically assigns ownership** during insertion.
- ✓ Users **cannot view/edit/delete books they do not own**.

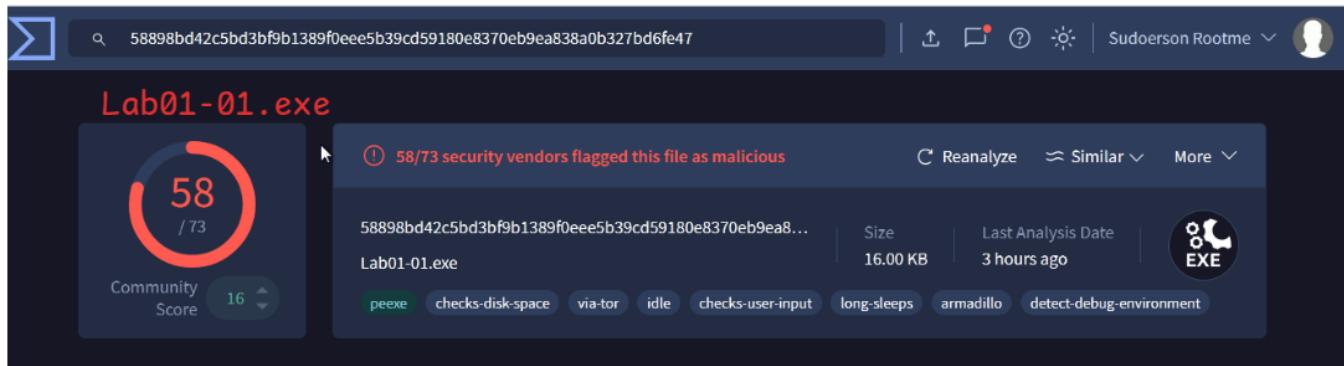
## Static Analysis - Lab 9

### Lab - 1-1

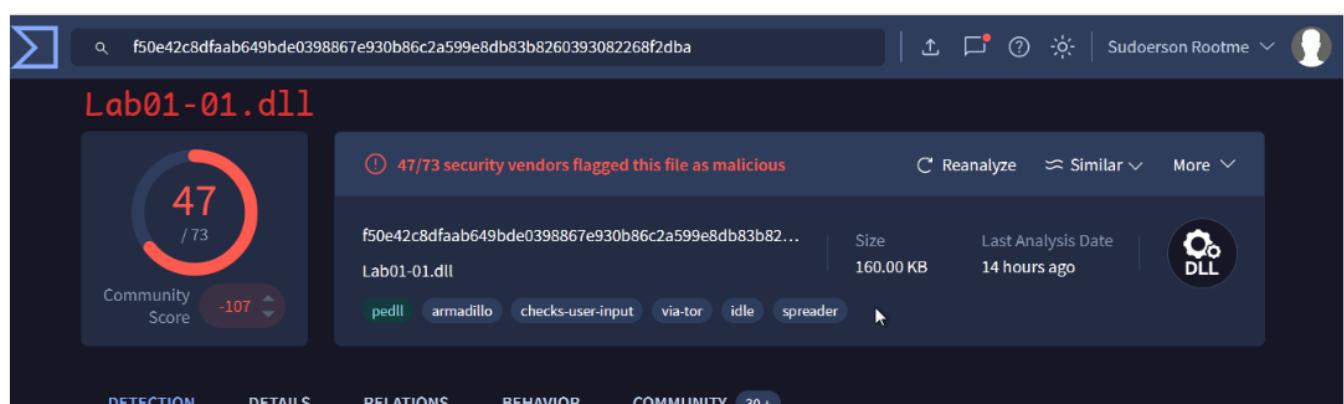
## Questions

1. Upload the files to <http://www.VirusTotal.com/> and view the reports. Does either file match any existing antivirus signatures?

## Answer



The screenshot shows the VirusTotal analysis page for the file `Lab01-01.exe`. The file has a community score of 58 out of 73, with 16 detections. The report indicates that 58/73 security vendors flagged the file as malicious. The file size is 16.00 KB and it was last analyzed 3 hours ago. The file type is EXE. Detection tags include `peexe`, `checks-disk-space`, `via-tor`, `idle`, `checks-user-input`, `long-sleeps`, `armadillo`, and `detect-debug-environment`.

The screenshot shows the VirusTotal analysis page for the file `Lab01-01.dll`. The file has a community score of 47 out of 73, with -107 detections. The report indicates that 47/73 security vendors flagged the file as malicious. The file size is 160.00 KB and it was last analyzed 14 hours ago. The file type is DLL. Detection tags include `pedll`, `armadillo`, `checks-user-input`, `via-tor`, `idle`, and `spreader`.

Yes, 58/73 vendors flagged .exe as malware Yes, 47/73 vendors flagged .dll as malware

2. When were these files compiled

## Answer

History ⓘ	
<b>Lab01-01.exe</b>	
Creation Time	2010-12-19 16:16:19 UTC
First Seen In The Wild	2012-01-08 02:19:06 UTC
First Submission	2012-02-16 07:31:54 UTC
Last Submission	2025-03-20 00:35:18 UTC
Last Analysis	2025-03-19 20:46:16 UTC

History ⓘ	
<b>Lab01-01.dll</b>	
Creation Time	2010-12-19 16:16:38 UTC
First Seen In The Wild	2010-12-19 09:16:38 UTC
First Submission	2011-07-04 19:57:48 UTC
Last Submission	2025-03-19 21:28:12 UTC
Last Analysis	2025-03-19 09:54:43 UTC

Both of them were compiled on 19 December, 2010

### 3. Is there any indicators that the files are packed or obfuscated?

#### Answer

PEview - C:\Users\Aparchit\Desktop\LabFiles\Practical Malware Analysis Labs\BinaryCollection\Chapter\_1L\Lab01-01.exe

File View Go Help

RVA	Data	Description	Value
00002000	00002124	Hint/Name RVA	001B CloseHandle
00002004	00002132	Hint/Name RVA	02B0 UnmapViewOfFile
00002008	00002144	Hint/Name RVA	01B5 IsBadReadPtr
0000200C	00002154	Hint/Name RVA	01D6 MapViewOfFile
00002010	00002164	Hint/Name RVA	0035 CreateFileMappingA
00002014	0000217A	Hint/Name RVA	0034 CreateFileA
00002018	00002188	Hint/Name RVA	0090 FindClose
0000201C	00002194	Hint/Name RVA	009D FindNextFileA
00002020	000021A4	Hint/Name RVA	0094 FindFirstFileA
00002024	000021B6	Hint/Name RVA	0028 CopyFileA
00002028	00000000	End of Imports	KERNEL32.dll
0000202C	000021D0	Hint/Name RVA	0291 malloc
00002030	000021DA	Hint/Name RVA	0249 exit
00002034	000021EE	Hint/Name RVA	00D3 _exit
00002038	000021F6	Hint/Name RVA	0048 _XcptFilter
0000203C	00002204	Hint/Name RVA	0064 __p__initenv
00002040	00002214	Hint/Name RVA	0058 __getmainargs
00002044	00002224	Hint/Name RVA	010F __initterm
00002048	00002230	Hint/Name RVA	0083 __setusermatherr
0000204C	00002244	Hint/Name RVA	009D __adjust_fdiv
00002050	00002254	Hint/Name RVA	006A __p__commode
00002054	00002264	Hint/Name RVA	006F __p__fmode
00002058	00002272	Hint/Name RVA	0081 __set_app_type
0000205C	00002284	Hint/Name RVA	00CA __except_handler3
00002060	00002298	Hint/Name RVA	00B7 __controlfp
00002064	000022A6	Hint/Name RVA	01C1 __strcmp
00002068	00000000	End of Imports	MSVCRT.dll

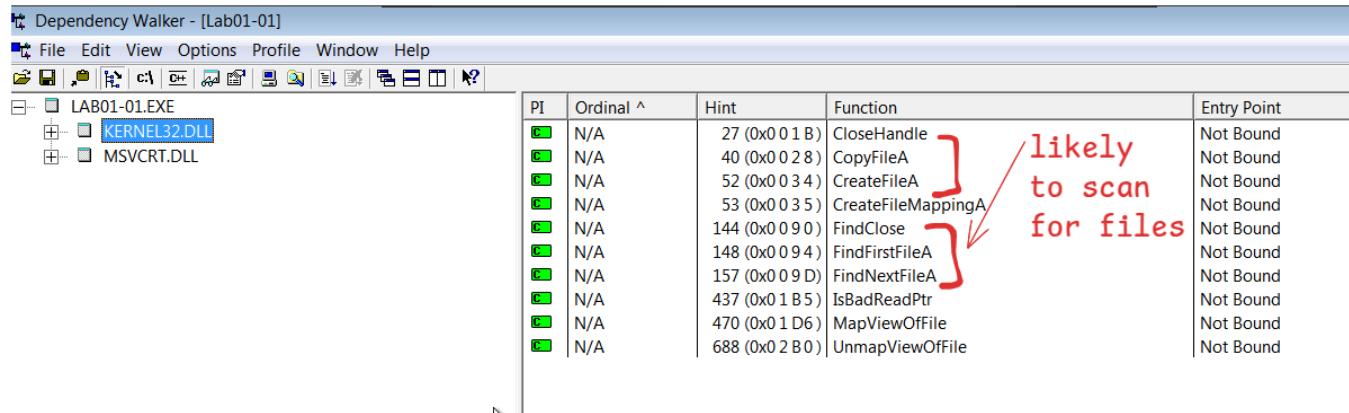
has well structured sections as normal PE file

all imports are shown

The file has well structured sections as well as standard imports. So, there is no sign of pack or obfuscation

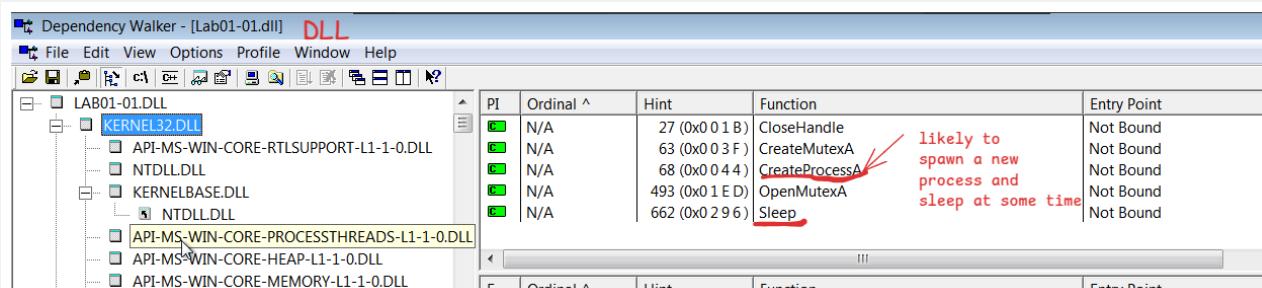
#### 4. Do any imports hint at what this malware does? If so, which imports are they?

##### Answer



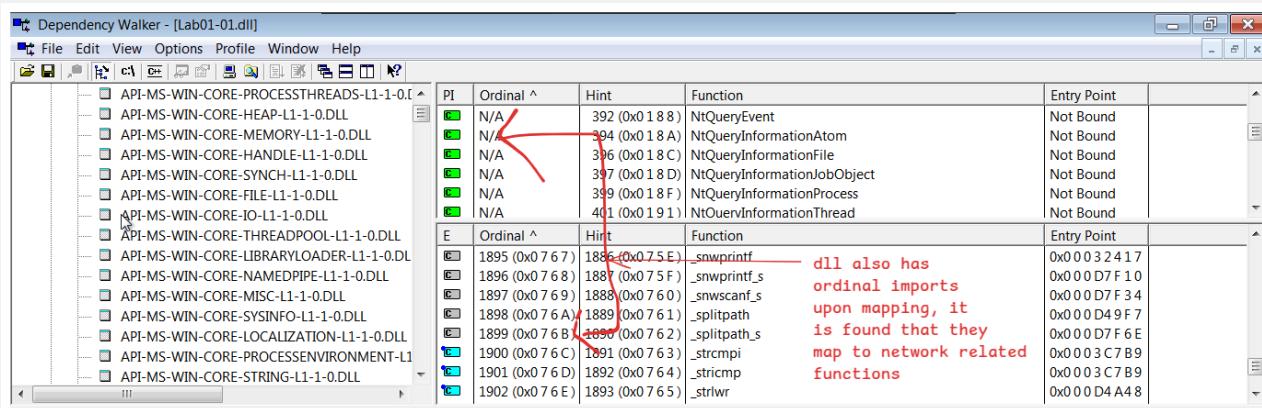
PI	Ordinal ^	Hint	Function	Entry Point
	N/A	27 (0x0 01 B)	CloseHandle	Not Bound
	N/A	40 (0x0 02 8)	CopyFileA	Not Bound
	N/A	52 (0x0 03 4)	CreateFileA	Not Bound
	N/A	53 (0x0 03 5)	CreateMappingA	Not Bound
	N/A	144 (0x0 09 0)	FindClose	Not Bound
	N/A	148 (0x0 09 4)	FindFirstFileA	Not Bound
	N/A	157 (0x0 09 D)	FindNextFileA	Not Bound
	N/A	437 (0x0 1B 5)	IsBadReadPtr	Not Bound
	N/A	470 (0x0 1D 6)	MapViewOfFile	Not Bound
	N/A	688 (0x0 2B 0)	UnmapViewOfFile	Not Bound

The imports in kernel32 contains file operations related functions. So, it must be looking scanning for files in the filesystem.



PI	Ordinal ^	Hint	Function	Entry Point
	N/A	27 (0x0 01 B)	CloseHandle	Not Bound
	N/A	63 (0x0 03 F)	CreateMutexA	Not Bound
	N/A	68 (0x0 04 4)	CreateProcessA	Not Bound
	N/A	493 (0x0 1 E D)	OpenMutexA	Not Bound
	N/A	662 (0x0 2 9 6)	Sleep	Not Bound

The dll is likely to spawn a new process and sleep at some time



PI	Ordinal ^	Hint	Function	Entry Point
	N/A	392 (0x0 1 8 8)	NtQueryEvent	Not Bound
	N/A	394 (0x0 1 8 A)	NtQueryInformationAtom	Not Bound
	N/A	396 (0x0 1 8 C)	NtQueryInformationFile	Not Bound
	N/A	397 (0x0 1 8 D)	NtQueryInformationJobObject	Not Bound
	N/A	399 (0x0 1 8 F)	NtQueryInformationProcess	Not Bound
	N/A	401 (0x0 1 9 1)	NtQueryInformationThread	Not Bound

E	Ordinal ^	Hint	Function	Entry Point
	1895 (0x0 7 6 7)	1886 (0x0 7 5 E)	_snprintf	0x0 00 32 41 7
	1896 (0x0 7 6 8)	1887 (0x0 7 5 F)	_snprintf_s	0x0 00 D7 F1 0
	1897 (0x0 7 6 9)	1888 (0x0 7 6 0)	_snwscanf_s	0x0 00 D7 F3 4
	1898 (0x0 7 6 A)	1889 (0x0 7 6 1)	_splitpath	0x0 00 D4 9 F7
	1899 (0x0 7 6 B)	1890 (0x0 7 6 2)	_splitpath_s	0x0 00 D7 F6 E
	1900 (0x0 7 6 C)	1891 (0x0 7 6 3)	_strcmpi	0x0 00 3C 78 9
	1901 (0x0 7 6 D)	1892 (0x0 7 6 4)	_strcmp	0x0 00 3C 7B 9
	1902 (0x0 7 6 E)	1893 (0x0 7 6 5)	_strlwr	0x0 00 D4 A4 8

The dll also ordinal imports, upon mapping they reveal functions from socket programming showing that the malware is likely to make network connection

#### 1. Are there any other files or host-based indicators that you could look for on infected systems?

##### Answer

```
_controlip  
_strcmp  
kernel32.dll  
kernel32.dll  
.exe  
C:\*  
C:\Windows\System32\kernel32.dll  
Kernel32.  
Lab01-01.dll  
C:\Windows\System32\Kernel32.dll  
WARNING_THIS_WILL_DESTROY_YOUR_MACHINE  
PS C:\Users\Aparichit\Desktop\StaticAnalysisTools\Strings> kernel not kernel
```

because of this it is likely malicious and we are able to use this to search for infected systems.

## 1. What network-based indicators could be used to find this malware on infected machines?

Answer

```
Y^j  
=X  
WUS  
WUS  
NWUS  
u7WPS  
u&WUS  
WUS  
_^[]  
%  
CloseHandle  
Sleep  
CreateProcessA  
CreateMutexA  
OpenMutexA  
KERNEL32.dll  
WS2_32.dll  
strcmp  
MSVCRT.dll  
free  
_initterm  
malloc  
_adjust_fdiv  
exec  
sleep  
hello  
127.26.152.13 It might be connecting to the C2 server or sending files to this IP  
SHDFHUF  
/0I0[0h0p0  
141G1[111  
1Y2a2g2r2  
3!3}3  
PS C:\Users\Aparichit\Desktop\StaticAnalysisTools\Strings>
```

It must be connecting to the Command and Control Server

## 7. What would you guess the purpose of these files

## Answer

Based on everything we've enumerated above, we would guess that the executable is used to run the DLL which acts as a backdoor or remote access trojan (RAT). Based on the imports it's possible the executable searches to see if C:\windows\system32\kerne132.dll exists, and if it doesn't it may attempt to copy the malicious DLL to C:\windows\system32\kerne132.dll which is used for persistence. Upon executing the DLL, it likely contacts a C2 server at 127.26.152.13.

---

## Lab - 1-2

### Lab 1-2

Analyze the file *Lab01-02.exe*.

### Questions

1. Upload the *Lab01-02.exe* file to <http://www.VirusTotal.com/>. Does it match any existing antivirus definitions?
2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.
3. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?
4. What host- or network-based indicators could be used to identify this malware on infected machines?

## Questions

1. **Upload the Lab01-02.exe file to <http://www.VirusTotal.com/>. Does it match any existing antivirus definitions?**

## Answer

The screenshot shows the VirusTotal analysis interface for the file c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6. The main summary indicates a Community Score of 60 / 73, with 60/73 security vendors flagged as malicious. The file is identified as Lab01-02.exe, has a size of 3.00 KB, and was last analyzed 5 hours ago. The file type is EXE. Below the summary, there are tabs for DETECTION, DETAILS, RELATIONS, BEHAVIOR, and COMMUNITY (30+). The DETECTION tab is selected, showing a large '60/73' score. Under 'Crowdsourced YARA rules', two matches are listed:

- ⚠️ Matches rule UPX from ruleset UPX at <https://github.com/kevoreilly/CAPEv2> by kevoreilly  
↳ UPX dump on OEP (original entry point) - 18 days ago
- ⚠️ Matches rule UPX from ruleset UPX at <https://github.com/kevoreilly/CAPEv2> by kevoreilly  
↳ UPX dump on OEP (original entry point) - 18 days ago

Yes, 60/73 vendors detects this file as malware

**2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.**

**Answer**

The image shows two side-by-side windows of the PEview debugger. Both windows have a title bar "PEview - C:\Users\Aparichit\Desktop\LabFiles\Lab01" and a menu bar "File View Go Help".

**Left Window (Normal Executable):**

- File structure: IMAGE\_DOS\_HEADER, MS-DOS Stub Program, IMAGE\_NT\_HEADERS, IMAGE\_SECTION\_HEADER .text, IMAGE\_SECTION\_HEADER .rdata, IMAGE\_SECTION\_HEADER .data, SECTION .text, SECTION .rdata, IMPORT Address Table, IMPORT Directory Table, **IMPORT Name Table**, IMPORT Hints/Names & DLL Names, SECTION .data.
- Annotations: A red arrow points from the text "text segment" to the ".text" section. Another red arrow points from the text "read only data segment" to the ".rdata" section.
- Label: "Normal" centered below the window.
- Status bar: "Viewing IMPORT Name Table"

**Right Window (Packed Executable):**

- File structure: IMAGE\_DOS\_HEADER, MS-DOS Stub Program, IMAGE\_NT\_HEADERS, Signature, IMAGE\_FILE\_HEADER, IMAGE\_OPTIONAL\_HEADER, IMAGE\_SECTION\_HEADER UPX0, IMAGE\_SECTION\_HEADER UPX1, IMAGE\_SECTION\_HEADER UPX2, SECTION UPX0, **SECTION UPX1**, SECTION UPX2, IMPORT Directory Table, IMPORT Address Table, IMPORT DLL Names, IMPORT Hints/Names.
- Annotations: A red box highlights the "SECTION UPX1" section. A red arrow points from the text "there is no .text section which contains the instructions to be executed. Also, the section upx0, upx1, upx2 shows that the program has been packed with UPX" to the "SECTION UPX1" section.
- Label: "Packed" centered below the window.
- Status bar: "Viewing SECTION UPX1"

The file is not having standard sections like .text, .rdata, etc. Also sections like upx0, upx1, upx2, shows that the executable has been packed with UPX

## Unpacking it using upx tool

```
PS C:\Users\Aparichit\Desktop\OtherTools\upx-5.0.0-win64\upx-5.0.0-win64> .\upx.exe -d 'C:\Users\Aparichit\Desktop\LabFiles\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L\Lab01-02.exe' -o unpacked.exe
                                         Ultimate Packer for eXecutables
                                         Copyright (C) 1996 - 2025
UPX 5.0.0      Markus Oberhumer, Laszlo Molnar & John Reiser   Feb 20th 2025

File size      Ratio      Format      Name
-----        -----      -----      -----
16384 <-      3072     18.75%    win32/pe    unpacked.exe

Unpacked 1 file.
PS C:\Users\Aparichit\Desktop\OtherTools\upx-5.0.0-win64\upx-5.0.0-win64> _
```

`upx -d /path/to/packedfile -o unpackedfile.exe`

### 3. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?

#### Answer

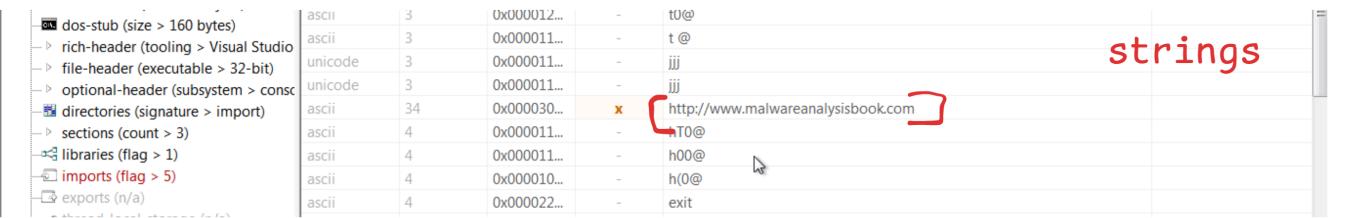
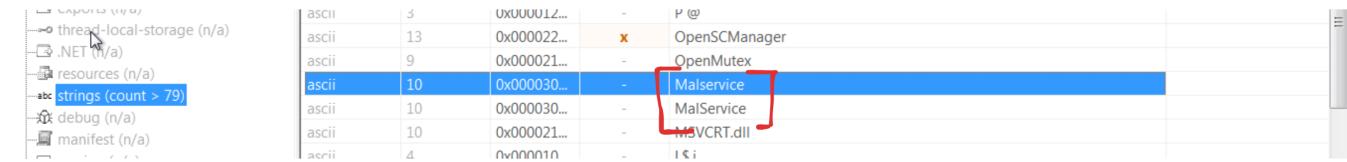
PI	Ordinal ^	Hint	Function	Entry Point
[ ]	N/A	0 (0x0 00)	InternetOpenUrlA	Not Bound
[ ]	N/A	0 (0x0 00)	InternetOpenA	Not Bound

PI	Ordinal ^	Hint	Function	Entry Point
[ ]	N/A	0 (0x0 00)	CreateServiceA	Not Bound
[ ]	N/A	0 (0x0 00)	StartServiceCtrlDispatcherA	Not Bound
[ ]	N/A	0 (0x0 00)	OpenSCManagerA	Not Bound

InternetOpenA, InternetOpenUrlA connects to the internet and CreateServiceA creates a service. It could be creating a process which will then be connecting to the internet.

### 4. What host- or network-based indicators could be used to identify this malware on infected machines?

#### Answer

dos-stub (size > 160 bytes)	ascii	3	0x000012...	-	tU@
rich-header (tooling > Visual Studio)	ascii	3	0x000011...	-	t @
file-header (executable > 32-bit)	unicode	3	0x000011...	-	jjj
optional-header (subsystem > console)	unicode	3	0x000011...	-	jjj
directories (signature > import)	ascii	34	0x000030...	x	http://www.malwareanalysisbook.com
sections (count > 3)	ascii	4	0x000011...	-	HT0@
libraries (flag > 1)	ascii	4	0x000011...	-	h00@
imports (flag > 5)	ascii	4	0x000010...	-	h0@
exports (n/a)	ascii	4	0x000022...	-	exit

capabilities (n/a)	ascii	3	0x000012...	-	P @
thread-local-storage (n/a)	ascii	13	0x000022...	x	OpenSCManager
.NET (n/a)	ascii	9	0x000021...	-	OpenMutex
resources (n/a)	ascii	10	0x000030...	x	Malservice
strings (count > 79)	ascii	10	0x000030...	-	MalService
debug (n/a)	ascii	10	0x000021...	-	MSVCRT.dll
manifest (n/a)	ascii	4	0x000010	-	I \$ i

Looking at the strings of this file shows 2 interesting elements, 'malservice' and 'http://www.malwareanalysisbook.com'. Based on this we can assume that searching hosts for the scheduled service called 'malservice' and looking at any hosts connecting to 'http://www.malwareanalysisbook.com' would serve as reliable host and network indicators.

## Lab 1-3

### Questions

- Upload the Lab01-03.exe file to <http://www.VirusTotal.com/>. Does it match any existing antivirus definitions?

### Answer

The screenshot shows the PEStudio interface with a VirusTotal report. The left pane displays the file structure of a .exe file, with a specific entry 'virustotal (score > 67/73)' highlighted by a red box. The right pane is a table titled 'VirusTotal report on PEStudio' showing the results from 73 vendors. The table has columns for vendor, score, date, and age. Most vendors flagged it as malicious (red), while a few like AVG and CMC were undetected.

vendor (73/73)	score (67/73)	date (dd.mm.yyyy)	age (da...)
ALYac	Gen:Variant.Grafter.968808	19.03.2025	2
APEX	Malicious	19.03.2025	2
AVG	Win32:Evo-gen [Trj]	19.03.2025	2
Acronis	undetected	28.03.2024	358
AhnLab-V3	Trojan/Win.Generic.R427327	19.03.2025	2
Alibaba	TrojanClicker:Win32/Tnega.79cba6...	27.05.2019	2125
Antiy-AVL	Trojan/Win32.SGeneric	19.03.2025	2
Arcabit	Trojan.Grafter.DEC868	19.03.2025	2
Avast	Win32:Evo-gen [Trj]	19.03.2025	2
Avira	TR/Clicker.lhuqy	19.03.2025	2
Baidu	Win32.Trojan-Clicker.Agent.z	18.03.2019	2195
BitDefender	Gen:Variant.Grafter.968808	19.03.2025	2
Bkav	W32.AIDetectMalware	19.03.2025	2
CAT-QuickHeal	Trojan.Ghajarava.1732885931113...	18.03.2025	3
CMC	undetected	19.03.2025	2
CTX	exe.trojan.generic	19.03.2025	2
ClamAV	Win.Malware.Emoneg-9937593-0	19.03.2025	2
CrowdStrike	win/malicious_confidence_100% (...	26.10.2023	512
Cylance	Unsafe	09.01.2025	71
Cynet	Malicious (score: 100)	19.03.2025	2

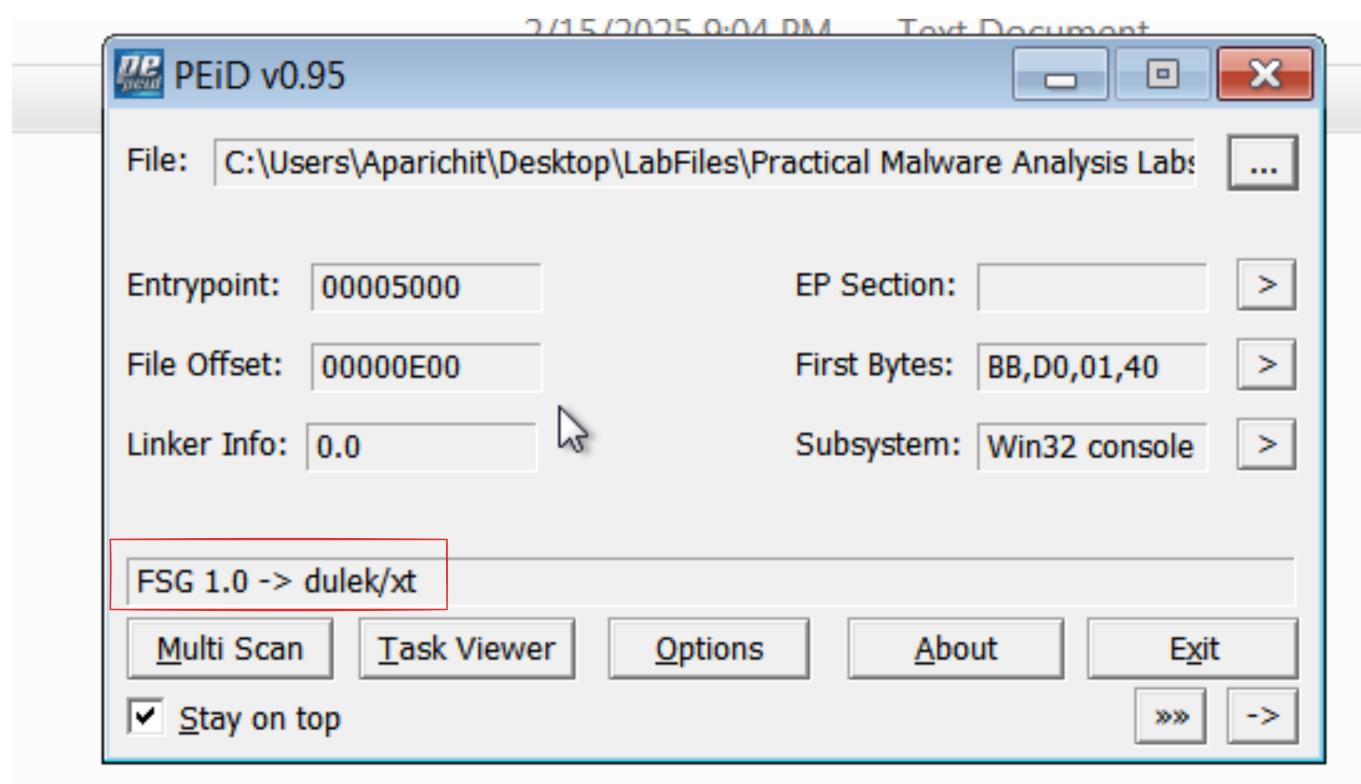
Yes, 67/73 vendors flagged the .exe as malware

## 2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.

### Answer

property	value	value	value
section	section[0]	section[1]	section[2]
name	n/a	n/a	n/a
section > sha256	n/a	E33C3C78CC31CD9CE...	52C905E29C1FD8BE94...
entropy	n/a	7.362	4.514
file > ratio (24.49%)	n/a	13.72 %	10.77 %
raw-address (begin)	0x00000000	0x00001000	0x00000E00
raw-address (end)	0x00000000	0x0000128C	0x00001000

PEStudio shows that the file has virtual size of 0x3000 but the raw size of 0x000. This is an indication of that the file might have been packed or obfuscated.



We can confirm it by using PEID tool which shows that the file is packed with FSG -> dulex/xt.

### 3. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?

#### Answer

can be answered deobfuscation

#### 4. What host- or network-based indicators could be used to identify this malware on infected machines?

##### Answer

can be answered deobfuscation

## Lab 1-4

### Questions

#### Lab 1-4

Analyze the file *Lab01-04.exe*.

#### **Questions**

1. Upload the *Lab01-04.exe* file to <http://www.VirusTotal.com/>. Does it match any existing antivirus definitions?
2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.
3. When was this program compiled?
4. Do any imports hint at this program's functionality? If so, which imports are they and what do they tell you?
5. What host- or network-based indicators could be used to identify this malware on infected machines?
6. This file has one resource in the resource section. Use Resource Hacker to examine that resource, and then use it to extract the resource. What can you learn from the resource?

28 Chapter 1

#### 1. Upload the files to <http://www.VirusTotal.com/> and view the reports. Does either file match any existing antivirus signatures?

##### Answer

vendor (72/72)	score (63/72)	date (dd.mm.yyyy)	age (days)
ALYac	Gen:Variant.Cerbu.64782	10.03.2025	11
APEX	Malicious	10.03.2025	11
AVG	Win32:DropperX-gen [Drp]	10.03.2025	11
Acronis	undetected	28.03.2024	358
AhnLab-V3	Trojan/Win.DownLoader.C5520690	11.03.2025	10
Alibaba	TrojanDownloader:Win32/Gofot.7...	27.05.2019	2125
Antiy-AVL	Trojan[Downloader]/Win32.AGene...	10.03.2025	11

Yes, 63/72 vendors flagged .exe as malware

## 2. Are there any indications that this file is packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it if possible.

### Answer

pestudio 9.60 - Malware Initial Assessment - www.winitor.com | c:\users\aparichit\Desktop\LabFiles\Practical Malware Analysis\Lab01-04.exe

File View Go Help

Imports (34) flag (9) type (1)

GetModuleHandleA implicit  
GetWindowsDirectoryA implicit  
MoveFileA implicit  
GetTempPathA implicit  
GetCurrentProcess implicit  
OpenProcess implicit  
CloseHandle implicit  
LoadResource implicit  
OpenProcessToken implicit  
LookupPrivilegeValueA implicit  
AdjustTokenPrivileges implicit  
snprintf implicit  
exit implicit  
XcptFilter implicit  
exit implicit  
p\_initenv implicit  
getmainargs implicit  
initterm implicit  
setusermatherr implicit  
adjust\_fdiv implicit  
p\_commode implicit  
p\_fmode implicit  
set app\_type implicit  
except\_handler3 implicit  
controlfn implicit

SECTION .text  
SECTION .rdata  
SECTION .data  
SECTION .rsrc

we can see clearly defined .text section .data section .rsrc section These are the most important sections of executables and are mostly packed

The file has clearly defined sections and headers. Also it has proper import table. This shows that the program is not packed or obfuscated.

## 3. When was this program compiled?

### Answer

strings (count > 167)  
debug (n/a)  
manifest (n/a)  
version (n/a)  
certificate (n/a)  
overlay (n/a)

	network-run-from-swap	0x0000	false
general			
stamp > compiler	0x5D69A2B3	Fri Aug 30 22:26:59 2019 (UTC)	
size	0x14	20 bytes	
file-header > location	0x000000EC - 0x000000...	0x000000EC - 0x00000100	
signature	0x00004550	PE00	
machine	0x014C	Intel-386	
sections > count	0x0004	4	
pointer-symbol-table	0x00000000	0x00000000	
number-of-symbols	0x00000000	0x00000000	

sha256 > OFA1498340FCA6C562CFA389AD3E93395F44C72FD128D7BA08579 cpu > 32-bit file > type > executable subsystem > GUI entry-point > 0x000015CF

The compiled time as shown is: Aug 2019. But this might be faked too.

#### 4. Do any imports hint about program functionality?

##### Answer

Dependency Walker - [Lab01-04]

File	Edit	View	Options	Profile	Window	Help
PI	Ordinal ^	Hint	Function		Entry Point	
LAB01-04.EXE	N/A	52 (0x0 3 4)	CreateFileA		Not Bound	
	N/A	70 (0x0 4 6)	CreateRemoteThread		Not Bound	
	N/A	163 (0x0 A 3)	FindResourceA		Not Bound	
	N/A	247 (0x0 F 7)	GetCurrentProcess		Not Bound	
	N/A	294 (0x0 1 2 6)	GetModuleHandleA		Not Bound	
	N/A	318 (0x0 1 3 E)	GetProcAddress		Not Bound	
	N/A	357 (0x0 1 6 5)	GetTempPathA		Not Bound	
	N/A	381 (0x0 1 7 D)	GetWindowsDirectoryA		Not Bound	
	N/A	450 (0x0 1 C 2)	LoadLibraryA		Not Bound	
	N/A	455 (0x0 1 C 7)	LoadResource		Not Bound	
	N/A	477 (0x0 1 DD)	MoveFileA		Not Bound	
	N/A	495 (0x0 1 EF)	OpenProcess		Not Bound	

most of  
these  
are file  
operations

Dependency Walker - [Lab01-04]

File	Edit	View	Options	Profile	Window	Help
PI	Ordinal ^	Hint	Function		Entry Point	
LAB01-04.EXE	N/A	247 (0x0 1 F 7)	GetCurrentProcess		Not Bound	
	N/A	294 (0x0 1 2 6)	GetModuleHandleA	findresourceA	Not Bound	
	N/A	318 (0x0 1 3 E)	GetProcAddress	loadResourceA	Not Bound	
	N/A	357 (0x0 1 6 5)	GetTempPathA	winexec	Not Bound	
	N/A	381 (0x0 1 7 D)	GetWindowsDirectoryA		Not Bound	
	N/A	450 (0x0 1 C 2)	LoadLibraryA		Not Bound	
	N/A	455 (0x0 1 C 7)	LoadResource		Not Bound	
	N/A	477 (0x0 1 DD)	MoveFileA		Not Bound	
	N/A	495 (0x0 1 EF)	OpenProcess		Not Bound	
	N/A	661 (0x0 2 9 5)	SizeofResource		Not Bound	
	N/A	723 (0x0 2 D 3)	WinExec		Not Bound	
	N/A	735 (0x0 2 DF)	WriteFile		Not Bound	

It might be  
loading some  
resource and  
executing it  
as a binary

This is suspicious because, we had seen exe file in the resource section in PEStudio

The files has imports for file operations but the imports like FindResourceA, LoadResourceA, winexec make it suspicious. Further importing .exe from ResourceHacker makes it more suspicious.

Dependency Walker - [Lab01-04]

File	Edit	View	Options	Profile	Window	Help
PI	Ordinal ^	Hint	Function		Entry Point	
LAB01-04.EXE	N/A	23 (0x0 0 1 7)	AdjustTokenPrivileges		Not Bound	
	N/A	245 (0x0 0 F 7)	LookupPrivilegeValueA		Not Bound	
	N/A	322 (0x0 1 4 2)	OpenProcessToken		Not Bound	

The imports from Advapi32 indicate that this is attempting to modify or change the token assigned to the execution of this process, presumably to elevate privileges or give extended access rights.

#### 5. What host- or network-based indicators could be used to identify this malware on infected machines?

##### Answer

Unexpectedly, the malware doesn't seem to have made any imports for network functions.

name	in...	signature (1)	location (from-to)	size (16384 bytes)	footprint (sha256)	entro...	language (
BIN	101	executable (cpu: 32-bit)	0x00004060 - 0x00008...	0x00004000 (163...	819B2DB1876D8584681179966...	0.852	English-US

But this resource is suspicious.

If we open the resource in resource hacker, we can find the executable. which was attached as a resource.

↳ exports (n/a)				
↳ thread-local-storage (n/a)				
↳ .NET (n/a)				
↳ resources (signature > executable)				
↳ strings (count > 167)				
↳ debug (n/a)				
↳ manifest (n/a)				

Also, the string tools also provides some network or host based indicator

**6. This file has one resource in the resource section. Use Resource Hacker to examine that resource, and then use it to extract the resource. What can you learn from the resource?**

## Answer

The screenshot shows the Dependency Walker interface. On the left, the DLL search order is listed: BIN101.EXE, KERNEL32.DLL, URLMON.DLL, and MSVCRT.DLL. URLMON.DLL is expanded, showing its internal exports. On the right, a table lists the exports of URLMON.DLL. The 'function' column contains the address of each export. A red box highlights the entry for URLDownloadToFileA.

PI	Ordinal ^	Hint	function	Entry Point
0	N/A	62 (0x0 3 E)	URLDownloadToFileA	Not Bound

The main executable had no imports for networking functions. All networking operations was being done by the attached resource. Also, the url in the strings suggests that the attached resource was downloading another malware for further exploitation.

# KFSensor Lab - Honeypot : Lab 10

# KF Sensor

## Initial Setup Info

```
Linux Host machine IP: 11.12.17.62
Windows VM IP : 11.12.2.44
Connected via bridged interface
```

## Pinging each other

```
example.sh
aparichit@SUSHANKYA:~/rough/syseclab/apparmor/class/data$ ping 11.12.2.44
PING 11.12.2.44 (11.12.2.44) 56(84) bytes of data.
64 bytes from 11.12.2.44: icmp_seq=1 ttl=128 time=0.447 ms      windows pinging
64 bytes from 11.12.2.44: icmp_seq=2 ttl=128 time=0.462 ms      Linux
^C
--- 11.12.2.44 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1036ms
rtt min/avg/max/mdev = 0.447/0.454/0.462/0.007 ms
aparichit@SUSHANKYA:~/rough/syseclab/apparmor/class/data$
```

```
PS C:\Users\Aparichit> ping 11.12.17.62
Pinging 11.12.17.62 with 32 bytes of data:
Reply from 11.12.17.62: bytes=32 time<1ms TTL=64      Linux pinging
Reply from 11.12.17.62: bytes=32 time<1ms TTL=64      windows
Reply from 11.12.17.62: bytes=32 time<1ms TTL=64
Reply from 11.12.17.62: bytes=32 time<1ms TTL=64
Ping statistics for 11.12.17.62:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Aparichit> ^X^X
```

Both the devices are connected to each other

## Initial port status of windows

```
(base) aparichit@SUSHANKYA:~/rough/syseclab/apparmor/class/data$ nmap 11.12.2.44
Starting Nmap 7.80 ( https://nmap.org ) at 2025-03-22 07:11 IST
Nmap scan report for 11.12.2.44
Host is up (0.00046s latency).
Not shown: 992 closed ports
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
1025/tcp   open  NFS-or-IIS
1026/tcp   open  LSA-or-nterm
1027/tcp   open  IIS
1028/tcp   open  unknown
1029/tcp   open  ms-lsa
```

Basic windows services

NOTE: firewall was turned off during this lab

Nmap done: 1 IP address (1 host up) scanned in 1.90 seconds

Basic windows services are open as the windows system firewall was turned off

## After Starting KF-Sensor

```
1029/tcp open  ms-tlsa
Nmap done: 1 IP address (1 host up) scanned in 1.90 seconds
(base) aparichit@SUSHANKYA:~/rough/syseclab/apparmor/class/data$ nmap 11.12.2.44
Starting Nmap 7.80 ( https://nmap.org ) at 2025-03-22 07:26 IST
Nmap scan report for 11.12.2.44
Host is up (0.0010s latency).
Not shown: 888 closed ports
PORT      STATE SERVICE
1/tcp      open  tcpmux
7/tcp      open  echo
9/tcp      open  discard
13/tcp     open  daytime
17/tcp     open  qotd
19/tcp     open  chargen
21/tcp     open  ftp
22/tcp     open  ssh
23/tcp     open  telnet
25/tcp     open  smtp
42/tcp     open  nameserver
53/tcp     open  domain
80/tcp     open  http
81/tcp     open  hosts2-ns
82/tcp     open  xfer
83/tcp     open  mit-ml-dev
110/tcp    open  pop3
111/tcp    open  rpcbind
113/tcp    open  ident
119/tcp    open  nntp
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
143/tcp    open  imap
443/tcp    open  https
445/tcp    open  microsoft-ds
465/tcp    open  smtps
548/tcp    open  afp
587/tcp    open  submission
```

After starting KFSensor we can see a lots of services open. This is exactly what a honey pot does.

## Logged nmap scan in Kfsensor

Recent Activity

my linux host

ID	Start	Durat...	Pr...	Sen...	Name	Visitor	Description
119	3/22/2025 7:26:09 AM.184	0.000	TCP	110	POP3	11.12.17.62	
118	3/22/2025 7:26:09 AM.184	0.000	TCP	111	sunrpc	11.12.17.62	
117	3/22/2025 7:26:09 AM.184	0.000	TCP	113	ident	11.12.17.62	
116	3/22/2025 7:26:09 AM.184	4.945	TCP	143	IMAP	11.12.17.62	
115	3/22/2025 7:26:09 AM.184	0.000	TCP	143	Multi-port ...	11.12.17.62	Multi-port Scan
114	3/22/2025 7:26:09 AM.184	0.000	TCP	443	IIS HTTPS	11.12.17.62	
113	3/22/2025 7:26:09 AM.184	0.000	TCP	587	SMTP TLS	11.12.17.62	
112	3/22/2025 7:26:09 AM.184	0.000	TCP	593	CIS	11.12.17.62	
111	3/22/2025 7:26:09 AM.184	0.000	TCP	993	IMAPS	11.12.17.62	
110	3/22/2025 7:26:09 AM.184	0.000	TCP	1723	Microsoft P...	11.12.17.62	
109	3/22/2025 7:26:09 AM.184	0.000	TCP	1801	MS MQS	11.12.17.62	
108	3/22/2025 7:26:09 AM.184	0.016	TCP	2103	MS MQS	11.12.17.62	
107	3/22/2025 7:26:09 AM.200	0.000	TCP	2869	MS UPNP H...	11.12.17.62	
106	3/22/2025 7:26:09 AM.184	0.000	TCP	995	POP3S	11.12.17.62	
105	3/22/2025 7:26:09 AM.200	0.000	TCP	2967	Symantec A...	11.12.17.62	
104	3/22/2025 7:26:09 AM.200	0.000	TCP	3283	Apple Net ...	11.12.17.62	
103	3/22/2025 7:26:09 AM.200	0.000	TCP	3306	MySQL Serv...	11.12.17.62	
102	3/22/2025 7:26:09 AM.200	0.000	TCP	4443	IIS HTTPS	11.12.17.62	
101	3/22/2025 7:26:09 AM.200	0.000	TCP	8080	IIS Proxy	11.12.17.62	
100	3/22/2025 7:26:09 AM.200	0.000	TCP	8084	IIS Proxy	11.12.17.62	
99	3/22/2025 7:26:08 AM.654	0.265	TCP	443	IIS HTTPS	11.12.17.62	
98	3/22/2025 7:26:09 AM.200	4.914	TCP	25	SMTP	11.12.17.62	
97	3/22/2025 7:26:09 AM.200	0.000	TCP	25	DOS Attack	11.12.17.62	DOS Attack
96	3/22/2025 7:26:09 AM.200	0.000	TCP	23	Telnet	11.12.17.62	

## Capturing dos attack using hping3

```
sudo hping3 -S -p 80 -c 100 11.12.2.44
```

DOS Attack

193	3/22/2025 7:41:47 AM.276	0.000	TCP	106..	IIS Proxy	11.12.17.62
192	3/22/2025 7:41:47 AM.276	4.851	TCP	491..	Vista svc	11.12.17.62
191	3/22/2025 7:41:47 AM.276	0.000	TCP	491..	DOS Attack	11.12.17.62
190	3/22/2025 7:41:47 AM.244	0.016	TCP	443	IIS HTTPS	11.12.17.62
189	3/22/2025 7:41:47 AM.260	0.000	TCP	465	SMTP SSL	11.12.17.62

generating large traffic using hping identified as DOS attack