**Amity University Rajasthan**

**Amity Institute of Information Technology**

**Master of Science – Cyber Security**

**CYBER SECURITY - I LAB**

| **COURSE CODE** | **L** | **T** | **P/FW** | **CREDIT UNITS** |
| --- | --- | --- | --- | --- |
| **MCS123** | **-** | **-** | **2** | **1** |

**Experiments**

1. Installation of LINUX Operating System
2. Implementation of general purpose utilities commands.
3. Implementation of user & session management commands.
4. Implementation of file system navigation commands, text processing tools, communication commands.
5. Write a shell script program to display “HELLO WORLD”.
6. Write a shell script program to develop a scientific calculator.
7. Write a shell script program that prompts the user for the password. The user has maximum of 3 attempts. If the user enters the correct password, the message “Correct Password” is displayed else the message “Wrong Password”.
8. Write a program to generate all combinations of 1, 2 and 3 using for loop.
9. Write a shell script program that will receive any number of filenames as arguments. The shell script should check whether such files already exist. If they do, then it should be reported. The files that do not exist should be created in a sub-directory called mydir. The shell script should first check whether the sub-directory mydir exists in the current directory. If it doesn’t exist, then it should be created. If mydir already exists, then it should be reported along with the number of files that are currently present in mydir.
10. Write a Shell Script program to show the implementation of positional parameters.
11. Write a grep/egrep shell script program to find the number of words character, words and lines in a file.
12. Shell Script program to implement read, write, and execute permissions.
13. Implement SQL DDL Commands (Create, Drop, Alter, Truncate)
14. Implement SQL DML Commands (Select)
15. Implement SQL DCL Commands (Grant, Revoke)
16. Implement SQL TCP Commands (Commit, Rollback)

**Examination Scheme:**

| **IA** | | | | **EE** | |
| --- | --- | --- | --- | --- | --- |
| **A** | **PR** | **LR** | **V** | **PR** | **V** |
| 5 | 15 | 15 | 15 | 25 | 25 |

Note: IA –Internal Assessment, EE- External Exam, PR- Performance, LR – Lab Record, V – Viva.

**1. Installation of LINUX Operating System**

* **Objective**: Install a Linux distribution like Ubuntu, Fedora, or CentOS.
* **Steps**:
  1. **Download ISO**: Obtain the ISO file for the desired Linux distribution from its official website.
  2. **Create Bootable USB**: Use a tool like Rufus (for Windows) or dd (for Linux) to create a bootable USB.
  3. **Boot from USB**: Restart your computer and boot from the USB drive by selecting it in the BIOS/UEFI settings.
  4. **Install Linux**: Follow the on-screen instructions to install Linux. Choose partitioning options, user setup, etc.
  5. **Post-Installation**: After installation, remove the USB and reboot. Log in to your new Linux system.

**2. Implementation of General Purpose Utilities Commands**

* **Objective**: Practice using basic Linux commands.
* **Example Commands**:
  + **ls**: Lists directory contents.
  + **cp**: Copies files.
  + **mv**: Moves or renames files.
  + **rm**: Deletes files.
  + **mkdir**: Creates directories.
  + **rmdir**: Removes directories.

**Example**:

bash

mkdir test\_directory # Creates a directory named test\_directory

ls # Lists files in the current directory

cp file.txt test\_directory/ # Copies file.txt to test\_directory

**3. Implementation of User & Session Management Commands**

* **Objective**: Learn to manage users and sessions.
* **Example Commands**:
  + **whoami**: Displays the current user.
  + **adduser or useradd**: Adds a new user.
  + **passwd**: Changes user password.
  + **su**: Switches to another user.
  + **logout**: Ends the session.

**Example**:

bash

sudo adduser newuser # Adds a new user named newuser

passwd newuser # Sets a password for newuser

su newuser # Switches to newuser

logout # Logs out from the current session

**4. Implementation of File System Navigation Commands, Text Processing Tools, Communication Commands**

* **Objective**: Navigate the file system, process text, and use communication commands.
* **Example Commands**:
  + **cd**: Changes directory.
  + **grep**: Searches for patterns in files.
  + **awk**: Text processing.
  + **mail**: Sends emails.

**Example**:

bash

cd /var/log # Navigates to /var/log directory

grep "error" syslog # Searches for "error" in syslog

awk '{print $1, $3}' file.txt # Prints the 1st and 3rd columns from file.txt

echo "Test message" | mail -s "Subject" user@example.com # Sends an email

**5. Write a Shell Script Program to Display “HELLO WORLD”**

* **Script**:

bash

#!/bin/bash

echo "HELLO WORLD"

* **Execution**:

bash

chmod +x hello.sh # Makes the script executable

./hello.sh # Runs the script

**6. Write a Shell Script Program to Develop a Scientific Calculator**

* **Script**:

bash

#!/bin/bash

echo "Enter first number:"

read a

echo "Enter second number:"

read b

echo "Choose operation: + - \* / ^ (power)"

read op

case $op in

+) result=$(echo "$a + $b" | bc);;

-) result=$(echo "$a - $b" | bc);;

\\*) result=$(echo "$a \* $b" | bc);;

/) result=$(echo "scale=2; $a / $b" | bc);;

^) result=$(echo "$a ^ $b" | bc);;

\*) echo "Invalid operation"; exit 1;;

esac

echo "Result: $result"

* **Execution**:

bash

chmod +x calc.sh

./calc.sh

**7. Write a Shell Script Program that Prompts the User for the Password**

* **Script**:

bash

#!/bin/bash

correct\_password="password123"

attempts=3

while [ $attempts -gt 0 ]; do

echo "Enter your password:"

read -s input\_password

if [ "$input\_password" == "$correct\_password" ]; then

echo "Correct Password"

exit 0

else

echo "Wrong Password"

attempts=$((attempts-1))

if [ $attempts -gt 0 ]; then

echo "You have $attempts attempts left"

fi

fi

done

echo "No more attempts left. Exiting."

exit 1

* **Execution**:

bash

chmod +x password\_check.sh

./password\_check.sh

**8. Write a Program to Generate All Combinations of 1, 2, and 3 Using for Loop**

* **Script**:

bash

#!/bin/bash

for i in 1 2 3; do

for j in 1 2 3; do

for k in 1 2 3; do

echo "$i $j $k"

done

done

done

* **Execution**:

bash

chmod +x combinations.sh

./combinations.sh

**9. Write a Shell Script Program to Manage Files in a Sub-Directory**

* **Script**:

bash

#!/bin/bash

if [ ! -d "mydir" ]; then

mkdir mydir

echo "Directory mydir created."

else

echo "Directory mydir already exists."

echo "Currently, mydir contains $(ls mydir | wc -l) files."

fi

for file in "$@"; do

if [ -e "$file" ]; then

echo "File $file already exists."

else

touch "mydir/$file"

echo "File $file created in mydir."

fi

done

* **Execution**:

bash

chmod +x manage\_files.sh

./manage\_files.sh file1 file2 file3

**10. Write a Shell Script Program to Show the Implementation of Positional Parameters**

* **Script**:

bash

#!/bin/bash

echo "First parameter: $1"

echo "Second parameter: $2"

echo "All parameters: $@"

echo "Number of parameters: $#"

* **Execution**:

bash

chmod +x positional\_parameters.sh

./positional\_parameters.sh arg1 arg2 arg3

**11. Write a grep/egrep Shell Script Program to Find the Number of Characters, Words, and Lines in a File**

* **Script**:

bash

#!/bin/bash

if [ -f "$1" ]; then

echo "File: $1"

echo "Characters: $(wc -m < $1)"

echo "Words: $(wc -w < $1)"

echo "Lines: $(wc -l < $1)"

else

echo "File not found!"

fi

* **Execution**:

bash

chmod +x file\_info.sh

./file\_info.sh filename.txt

**12. Shell Script Program to Implement Read, Write, and Execute Permissions**

* **Script**:

bash

#!/bin/bash

echo "Enter filename:"

read filename

echo "Choose permission: r(read), w(write), x(execute)"

read permission

case $permission in

r) chmod +r $filename ;;

w) chmod +w $filename ;;

x) chmod +x $filename ;;

\*) echo "Invalid permission"; exit 1 ;;

esac

echo "Permission set successfully."

* **Execution**:

bash

chmod +x set\_permissions.sh

./set\_permissions.sh

**13. Implement SQL DDL Commands (Create, Drop, Alter, Truncate)**

* **Objective**: Practice basic SQL Data Definition Language commands.
  + **Create**:

sql

CREATE TABLE students (

id INT PRIMARY KEY,

name VARCHAR(100),

age INT

);

* + **Drop**:

sql

DROP TABLE students;

* + **Alter**:

sql

ALTER TABLE students ADD COLUMN address VARCHAR(255);

* + **Truncate**:

sql

TRUNCATE TABLE students;