You have to perform this activity on your laptops using your Hacking Lab environment. It is open notes, and open chatGPT activity. Please do your own work. Any cheating activity from your classmate's will award a zero to both the students.

This activity is about operating system, networking, and cryptography concepts implementation. You need to use Kali Linux (kali) and Ubuntu Server (us) and Metasploitable2 (m2) to execute relevant commands, and understand the concepts. Keep a note of the sequence of commands you used, your observations, and any comment if you want to make. Get ready for the viva.

Task 01:

a) On the ubuntu server terminal, write down the command to check what all services that are managed by xinetd server. Give commands to ensure that the xinetd server should automatically start whenever the system gets booted. Write the commands in your notebook.

b) M2 has a weird kind of display, i.e., a small screen size and you cannot scroll up or down. It is a good idea to get a ssh login of M2 in kali. Open a new terminal window on Kali and login to M2 via ssh as user msfadmin. Write the commands in your notebook.

c) Run Apache web server on ubuntu machine. Create a file named evil_file.txt inside the ubuntu server at the appropriate location, having text "You are hacked". Can you download it on the Kali machine using the curl utility? Mention all the steps and commands to do this task.

d) Run vsftpd server on ubuntu machine. Login from kali on ubuntu server using ftp client as anonymous user. See in which ubuntu server directory this user is jailed. Try to upload and download files using put/mput and get/mget commands. Mention all the steps and commands to do this task.

e) Give a command that will display the arp table of your machine. If it does not contain the ip-mac address mapping of all the machines, use some command that will do that.

f) Give a set of commands that will use the dig command, to iteratively send requests to get the IP of www.arifbutt.me. Starting from the root server down the DNS hierarchy.

Task 02:

On Kali, start wireshark and set its filter to capture the telnet packets only. Now from Kali do a remote login on ubuntu server using telnet service. See if the password is transmitted in clear and if you can capture it. Repeat the same for remote login service ssh. What is the difference?

Task 03:

Create a user bob on Kali Linux machine, and a user alice on ubuntu server machine. Create a private-public key pair for both users using openssl command, and the Data Encryption Standard (DES) algorithm. Now use scp command to copy bob's public key from kali to ubuntu server and alice's public key from ubuntu server to kali. When both the users have their public-private key pairs as well as the other user's public key, they are ready to exchange cipher messages by digitally signing them. Suppose Bob wants to send a secret message to Alice (from kali to ubuntu server) by encrypting the message as well as digitally signing it to avoid

impersonation or masquerading. Mention the commands that Bob will perform to do this task. On the receiver side (ubuntu server), when Alice receives the message, she will decipher the message as well as verify that it is signed by Bob. Mention the commands that Alice will perform to do this task.

Task 04:

1. Create a simple login.html file that displays a simple login as shown. Place the file in your ubuntu server machine in Login Form the /var/www/html/ directory with Apache2 service running at port 80 and 443. On Kali, inside a browser give the address of the ubuntu server mentioning the index1.html and see if you can access the login form on kali Linux. Note, when you click the Login button, nothing happens R

2. On ubuntu server machine, start the mysql server and locally login to the server (you should know the root user and his/her password on your mysql server running on the ubuntu server). Once logged in, create a new database named mydb with a single table named users inside it with a username column and a password column. Use the INSERT statement and add three users in that table (storing hashed passwords using password_hash () in PHP). Once done, verify using sql commands on your ubuntu server terminal.

3. Now, you need to update the login.html file created in step-1 on ubuntu server and add some php code in it. So let us first make a copy of it with the name of login.php file. Add appropriate php code that will make a connection with the database (mydb) by specifying the ubuntu server IP, database user, database password, and database name that you have created in previous step. The php code will receive the username and password entered by a user on the login form and compare them with the existing user's names and hashed passwords inside the database. If a match exists, a login success message will be sent back from the web server to the browser, otherwise a login failure message will be sent.

4. Start wireshark on Kali, and then open the browser and access the login.php file from the web server running on ubuntu machine. When a user submits the login form, the browser sends an HTTP POST request to the server. The form data, including the username and password is included in the request body in plaintext. So, any attacker on the same network can intercept and read the credentials. Make your hands dirty to practically see a proof of this concept.

5. To mitigate/avoid the MitM attack in case of http, let us now try to access the login form using https instead of http. For this first of all you have to generate a self-signed Digital certificate (SSL certificate). On the server side you have to copy the certificate file inside the /etc/ssl/certs/ directory and the private key inside the /etc/ssl/private/ directory and then edit the /etc/apache2/sites-available/default-ssl.conf configuration file appropriately. Finally, enable ssl and restart apache service on ubuntu server machine. On the Kali Linux machine, you need to import the self-signed certificate inside your browser settings and mark it as trusted. After you have done this, you need to access the login.php file using https