**List of Practical**

1. Use Wireshark tool and explore the packet format and content at each OSI layer.
2. Study of following commands and tools:
3. IFconfig/IPconfig
4. Whois
5. Nslookup
6. PING
7. Traceroute
8. Telnet
9. Secure Shell
10. Tcpdump and Windump
11. Create a malicious program: Virus
12. TCP / UDP connectivity using Netcat
13. TCP scanning using NMAP.
14. Port scanning using NMAP.
15. Exploit Web Application Security using DVWA.
16. Examine SQL injection attack.
17. Study of the features of firewall in providing network security and to set Firewall Security in windows.
18. Study of different types of vulnerabilities for hacking a websites /Web Applications.
19. Wireshark tutorial for beginner – network scanner and analyzer – open source tool
20. Use Wireshark tool and explore the packet format and content at each OSI layer.
21. Study of the features of firewall in providing network security and to set Firewall Security in windows.
22. Steps to ensure Security of any one web browser (Mozilla Firefox/Google Chrome)
23. Study of different types of vulnerabilities for hacking a websites /Web Applications.
24. Analysis the Security Vulnerabilities of E-commerce services.
25. Examine SQL injection attack.
26. Analysis the security vulnerabilities of E-Mail Application
27. Explore following tools

Nmap

NetCat

1. Evaluate network defense tools for following

(i) IP spoofing

(ii) DOS attack

**Open Source Software/learning website:**

[www.wireshark.org](http://www.wireshark.org)

**IFconfig/IPconfig, Whois, Nslookup, PING, Traceroute, Telnet, Secure Shell,** Monitoring Tools and Software: Nagios, SolarWinds, Microsoft Network Monitor, **Wireshark**, **Snort**, Nmap, Nikto, OpenVAS, Metasploit, The Browser Exploitation Framework.

Explore the Nmap tool and list how it can be used for network defence.

1. Study of following network emulators:

a) WHOIS Search

b) Whois CLI Command

c) Nslookup

d) Host

e) Ping

f) Traceroute

g) Netstat

h) Tcpdump and Windump

2. Create a malicious program: Virus

3. TCP / UDP connectivity using Netcat

4. TCP scanning using NMAP.

5. Port scanning using NMAP.

6. TCP / UDP connectivity using Netcat .

**Part II - Exploits**

7. Exploit Web Application Security using DVWA.

Command Execution

 SQL Injection

 SQL Injection (Blind)

 File Inclusion

 File Upload

 Insecure CAPTCHA

 Brute Force

 CSRF

 Weak Session IDs

 XSS (DOM)

 XSS (Reflected)

 XSS (Stored)

 CSP Bypass

 JavaScript

8. Exploit Web application Security using DVWA

Automated SQL injection with SqlMap .

**List of practicals - https://gtupractical15.blogspot.com/p/cyber-security\_30.html**

[TCP scanning using NMAP](https://gtupractical15.blogspot.com/2019/06/tcp-scanning-using-nmap.html)

[Port scanning using NMAP](https://gtupractical15.blogspot.com/2019/06/port-scanning-using-nmap.html)

[TCP / UDP connectivity using Netcat](https://gtupractical15.blogspot.com/2019/06/tcp-udp-connectivity-using-netcat.html)

[How to install OpenVAS in Linux System ?](https://gtupractical15.blogspot.com/2019/06/how-to-install-openvas-in-linux-system.html)

[Network vulnerability using OpenVAS](https://gtupractical15.blogspot.com/2019/07/network-vulnerability-using-openvas.html)

[How to install DVWA in Linux System / Windows ?](https://gtupractical15.blogspot.com/2019/07/how-to-install-dvwa-in-linux-system.html)

Web application testing using DVWA

Manual SQL injection using DVWA

XSS using DVWA

Automated SQL injection with SqlMap

Cyber security practicals

<https://www.youtube.com/playlist?list=PLFoLtJ0skbALtIyPFdeSsCQGWIekOmgXN>

<https://www.studocu.com/in/document/rajasthan-technical-university-kota/cyber-security-lab/cybersecurity-lab-maual/12464448>

https://www.studocu.com/in/document/rajasthan-technical-university-kota/cyber-security-lab/cybersecurity-lab-maual/12464448

GTU syllabus practicals

1. Install Kali Linux. Examine the utilities and tools available in Kali Linux and find out which tool is the best for finding cyber attack/vulnerability.

2. Evaluate network defense tools for following

(i) IP spoofing

(ii) DOS attack

3. Explore the Nmap tool and list how it can be used for network defence.

4. Explore the NetCat tool.

5. Use Wireshark tool and explore the packet format and content at each OSI layer.

6. Examine SQL injection attack.

7. Perform SQL injection with SQLMap on vulnerable website found using google dorks.

8. Examine software keyloggers and hardware keyloggers.

9. Perform online attacks and offline attacks of password cracking.

10. Consider a case study of cyber crime, where the attacker has performed on line credit card fraud. Prepare a report and also list the laws that will be implemented on attacker..

List of practicals.

Opensource tools

Wireshark

openVAs

kali linux

Nmap –tool for port scanning

1. Wireshark tutorial for beginner – network scanner and analyzer – open source tool

Study and use the Wireshark for the various network protocols.

Installation of wireshark and observe the data transferred in client and server communication.

https://www.studocu.com/in/document/rajasthan-technical-university-kota/cyber-security-lab/cybersecurity-lab-maual/12464448

1. Implement the Brute Force Attack.

https://www.studocu.com/in/document/rajasthan-technical-university-kota/cyber-security-lab/cybersecurity-lab-maual/12464448

1. Explore following tools

Nmap

NetCat

Study of different wireless network components and features of any one of the Mobile Security Apps.

Study of the features of firewall in providing network security and to set Firewall Security in windows.

Steps to ensure Security of any one web browser (Mozilla Firefox/Google Chrome)

Study of different types of vulnerabilities for hacking a websites / Web Applications.

Analysis the Security Vulnerabilities of E-commerce services.

Analysis the security vulnerabilities of E-Mail Application  
|

3. /\*Change "(" to "<" and change ")" to ">" \*/
4. #include (iostream)
5. #include (string)
6. using namespace std;
7. /\*Prototypes\*/
8. void checkPassword(string password);
9. void recurse(int width, int position, string baseString);
10. /\*Global Variables\*/
11. char
12. chars[]={'z','y','x','w','v','u','t','s','r','q','p','o','n','m','l','k','j','i','h','g','f','e','d','c','b','a','9','8','7','6','5','4','3','2','1','0'};
13. string t;
14. /\*This function generates the password\*/
15. void recurse(int width, int position, string baseString)
16. {
17. for(int i=0;i<35;i++)
18. {
19. if (position < width-1)
20. {
21. recurse(width, position + 1, baseString+chars[i]);
22. }
23. checkPassword(baseString+chars[i]);
24. }
25. }
26. /\*This function checks to see if the generated password is correct\*/
27. void checkPassword(string password)
28. {
29. cout << "Trying this password: " << password << endl;
30. if (password==t) {
31. cout << "match [" << password << "]" << endl;
32. int pause;
33. cin >> pause;
34. exit(1);
35. }
36. }
37. int main()
38. {
39. cout << "Enter a string (No more then 10 characters for demonstration purposes): " << endl;
40. cin >> t;
41. int maxChars = 10;
42. for(int i = maxChars; i >0; i++)
43. {
44. cout << "Checking passwords width [" << i << "]..." << endl;
45. recurse(i,0,"");
46. }
47. return 0;
48. }

51. /\*Change "(" to "<" and change ")" to ">" \*/
52. #include (iostream)
53. #include (string)
54. using namespace std;
55. /\*Prototypes\*/
56. void checkPassword(string password);
57. void recurse(int width, int position, string baseString);
58. /\*Global Variables\*/
59. char
60. chars[]={'z','y','x','w','v','u','t','s','r','q','p','o','n','m','l','k','j','i','h','g','f','e','d','c','b','a','9','8','7','6','5','4','3','2','1','0'};
61. string t;
62. /\*This function generates the password\*/
63. void recurse(int width, int position, string baseString)
64. {
65. for(int i=0;i<35;i++)
66. {
67. if (position < width-1)
68. {
69. recurse(width, position + 1, baseString+chars[i]);
70. }
71. checkPassword(baseString+chars[i]);
72. }
73. }
74. /\*This function checks to see if the generated password is correct\*/
75. void checkPassword(string password)
76. {
77. cout << "Trying this password: " << password << endl;
78. if (password==t) {
79. cout << "match [" << password << "]" << endl;
80. int pause;
81. cin >> pause;
82. exit(1);
83. }
84. }
85. int main()
86. {
87. cout << "Enter a string (No more then 10 characters for demonstration purposes): " << endl;
88. cin >> t;
89. int maxChars = 10;
90. for(int i = maxChars; i >0; i++)
91. {
92. cout << "Checking passwords width [" << i << "]..." << endl;
93. recurse(i,0,"");
94. }
95. return 0;
96. }