

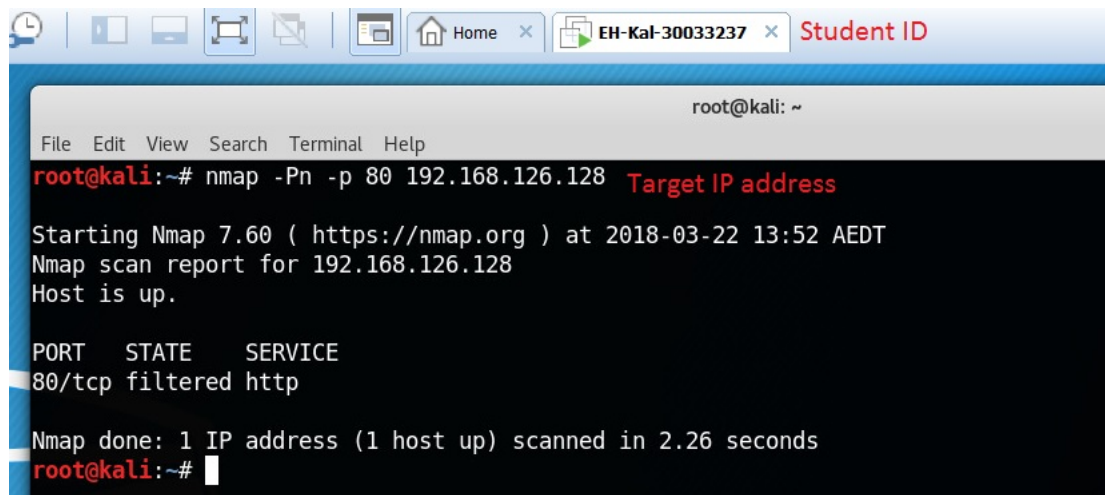
Pentesting Project

Due: 8pm, Friday 8 June 2018

In this project, you will pentest the Metasploitable2 VM according to the tasks described below. The tasks in this document will be a little harder than what you have seen in our lectures and labs. However, the basic skills involved will be similar.

Since pentesting is of exploration nature, you should try to complete the tasks by yourself without seeking help from tutors. There are hints and notes provided within this document to help you. Besides these, you should do research yourself first if you encounter difficulty in completing a task. For instance, if you do not understand what the 'xxx' command and its options do for you, use 'man xxx' to find out. After you have tried very hard and still cannot figure out, limited help can be obtained from tutors.

Write your answers for all tasks into a project report. When asked to grab a screenshot in a task below, the screenshot must include the top tab of the VM window that shows your Student ID. An exemplar screenshot is included as follows. If you are using VMs created on your own laptop, then the screenshot must show the IP address of the target somewhere. For instance, the target IP can appear in your command line, or if the command line does not include the target IP, you can use 'ip a' command to display the IP address intentionally.



```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap -Pn -p 80 192.168.126.128 Target IP address  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2018-03-22 13:52 AEDT  
Nmap scan report for 192.168.126.128  
Host is up.  
  
PORT      STATE      SERVICE  
80/tcp    filtered  http  
  
Nmap done: 1 IP address (1 host up) scanned in 2.26 seconds  
root@kali:~#
```

You are suggested to read the entire specification first, and then start with the tasks that are already covered by our lectures.

1 Port Scanning [2 marks]

1.1 Use nmap to conduct a default TCP ports scanning on Metasploitable2. [1 mark]

- According to the nmap output, how many TCP ports are open? (i.e., count the number of open ports and include it in the report)
- Also, you should include a screenshot showing the command line used and the first 10 lines of output.

1.2 Use nmap to scan all TCP ports of Metasploitable2. [1 mark]

- How many TCP ports are open this time?
- Also, you should include a screenshot showing the command line used and the first 10 lines of output.

2 Service and Vulnerability Detection [2 marks]

- 2.1 As seen from Task 1, the TCP port 8787 is open. Suppose you are interested in knowing which service is running on this TCP port. Use nmap to detect this. [1 mark]
- Include a screenshot showing your command line and the relevant output.
 - Then, based on the output, use your own words to describe the detected service and software version into your report.
- 2.2 In OpenVAS, create a target for Metasploitable2 with all TCP ports to be scanned. Then, create a task to scan this target with default configuration. [1 mark]
- Detail your steps into your project report and include a screenshot for target creation and task creation respectively.
 - According to the PDF report downloaded from this scan, how many results have the severity level of 'high'?

3 Exploitation [7 marks]

- 3.1 The "Metasploitable 2 Exploitability Guide" (<https://metasploit.help.rapid7.com/docs/metasploitable-2-exploitability-guide>) gives a great tutorial on how to exploit the Metasploitable2 VM. Please read through this guide, and especially focus on the 'Services: Backdoors' section. Then, accomplish the following tasks.
- The 'Services: Backdoors' section first describes how to manually exploit the backdoor in the tampered FTP server VSFTPD v2.3.4. Follow it to complete the exploitation on your Metasploitable2 using 'telnet'. Detail your steps and include a screenshot on your success. This screenshot should include the 'telnet' command line, and the results of executing the following commands after gaining access: 'id', 'ip addr show dev eth0', and 'hostname'. [1 mark]

Hints:

- You can also watch the following video on Youtube to get a clearer idea on this exploitation: <https://www.youtube.com/watch?v=8HONwJHDTtw>
 - 'Escape character is ^]' means that you need to enter Ctrl+] to exit telnet.
 - If you encounter the 'command not found' error with 'telnet', check whether you have included a semicolon in the end of your commands as illustrated in this tutorial. Alternatively, you can use 'netcat' instead. The 'netcat' will be covered in Week 7's lecture.
- The 'Services: Backdoors' section also describes how to exploit the old standby "ingreslock" backdoor that is listening on port 1524. Use the 'netcat' tool instead of 'telnet' to accomplish this exploitation. Detail your steps and include a screenshot on your success. This screenshot should include the 'netcat' command line, and the results of executing the following commands after gaining access: 'whoami', 'ip a show dev eth0', and 'pwd'. [1 mark]
- 3.2 The OpenVAS report for Metasploitable2 shows the 'Java RMI Server Insecure Default Configuration Remote Code Execution Vulnerability' on TCP port 1099. Follow the 'GNU Classpath RMI Registry' section in the following blog <https://tehaorum.wordpress.com/2015/06/14/metasploitable-2-walkthrough-an-exploitation-guide/> to exploit this vuln. Detail your steps and include a screenshot on your success. This screenshot should include the results of executing the following commands in

meterpreter session: `getuid` and `ifconfig`. [2 marks]

Hint: You should set payload and options as shown in the blog before executing 'exploit' or 'run'. Also, after you execute the 'run' or 'exploit' command, you may see some error messages. However, as long as you see the "Meterpreter session *n* opened ..." message, your exploitation is successful. In case you do not see the "meterpreter >" prompt after the 'exploit' command, enter the 'sessions' command. Then, you will see the meterpreter session *n* is listed, and you can use the 'sessions -i *n*' to enter that meterpreter session.

3.3 The OpenVAS report for Metasploitable2 also shows the 'Distributed Ruby (dRuby/DRb) Multiple Remote Code Execution Vulnerabilities' on TCP port 8787. Follow the following tutorial <http://www.hackingtutorials.org/metasploit-tutorials/hacking-druby-rmi-server-1-8/> to exploit this vuln. Detail your steps and include a screenshot on your success. This screenshot should include the results of executing the following commands after gaining access: 'id' and 'ip a show dev eth0'. [2 marks]

Hints:

- If you fail by following this tutorial, it is because the '17058.rb' needed for this exploitation is actually different from the 'drb_remote_codeexec.rb' in the current MSF installation. What you can do is: backup the current 'drb_remote_codeexec.rb' in MSF installation with a different name; copy '17058.rb' to be the new 'drb_remote_codeexec.rb'. Then, your exploitation will be successful.
- Follow the directory structure of MSF described in Lecture 5 slides to locate 'drb_remote_codeexec.rb'.
- The dRuby service is very fragile. When attacked, it may shutdown. If you have tried several times and still cannot succeed in exploitation, you need to check whether this service is still available by executing 'ss -antp | grep 8787'. If no port is listening on port 8787, you need to restart the Metasploitable2 to get dRuby running again.

3.4 The OpenVAS report for Metasploitable2 also shows the 'distcc Remote Code Execution Vulnerability' on TCP port 3632. Follow the Section 6 Steps 1-5 from the following tutorial https://www.computersecuritystudent.com/SECURITY_TOOLS/METASPLOITABLE/EXPLOIT/lesson2/index.html to exploit this vuln. Detail your steps and include a screenshot on your success. This screenshot should include the results of executing the following commands after gaining access: `whoami` and 'ip a show dev eth0'. [1 mark]

Note: The 'BackTrack' mentioned in this tutorial is the previous name of Kali Linux.

4 Post Exploitation [4 marks]

4.1 After completing Task 3.3, you obtain a command shell session with root privilege. In this post exploitation task, you will continue to gather some system information about the target using the root session obtained. Specifically, you should follow the "Post exploitation information gathering" section in the tutorial mentioned in Task 3.3 to complete this task.

Upon the completion of this task, you will see several text files generated under the "/root/.msf4/loot/" folder at Kali machine. One of these text files contains all of the user account names on Metasploitable2. In your report, detail your steps for this post exploitation, and include a screenshot showing the content of the file containing usernames. [1 mark]

Hint: You should use "use post/linux/gather/enum_system" instead of "use enum_system".

4.2 After completing Task 3.4, you will notice that the user account you get is 'daemon', not 'root'. Follow the Section 6 Steps 6-10 from the tutorial mentioned in Task 3.4 to escalate the privilege to 'root'. Detail your steps and include a screenshot on your success. This screenshot should include the results of

executing the following commands in the 'netcat' session: whoami and 'ip a show dev eth0'. The different things you should do from this tutorial are mentioned in the hints below. [3 marks]

Hints:

- Since in our university network, you are not allowed to visit exploit-db.com, you should obtain the 'exploit-8572.c' through another method. For example, you can use 'searchsploit' to find it in the local installation of exploit-db in Kali. You will see that it is named '8572.c' in the local installation of exploit-db.
- To upload '8572.c' to Metasploitable2, there can be several methods. Here we suggest the following two to you:
 - a) Use netcat, which is installed in both Metasploitable2 and Kali.
 - b) Start the Apache web server in Kali, and make '8572.c' downloadable through this web server.
- You can also watch the following video on Youtube to get a clearer idea on this privilege escalation: <https://www.youtube.com/watch?v=DoUZFHwZntY> .
- [Caution] The tutorial about this privilege escalation at <https://null-byte.wonderhowto.com/how-to/hack-metasploitable-2-including-privilege-escalation-0170603/> is wrong, as the root shell obtained by following this tutorial is on Kali, not on Metasploitable2.

5 Web Pentesting [5 marks]

In our lectures and labs, we used the DVWA as our web pentesting target. In this project, you will be asked to pentest another intentionally vulnerable web application called 'Mutillidae', which is also installed in Metasploitable2.

Before pentesting Mutillidae, you need to make one small change to the Mutillidae configuration file: /var/www/mutillidae/config.inc. In this file, there is the following line:

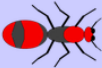
```
$dbname = 'metasploit';
```

You need to change it to:

```
$dbname = 'owasp10';
```

You should log into Metasploitable2 to make this change. You should 'cd /var/www/mutillidae' first, and then edit 'config.inc' with any text editor you prefer. For instance, if you prefer to use 'nano', then the command line you should use is 'sudo nano config.php'. Note that 'sudo' is the Linux command to allow selected non-root users to execute a command with root privilege. And such a selected non-root user can type his/her own password to pass the authentication when he/she uses 'sudo' to execute a command.

The use of Mutillidae is straightforward. Simply enter the following URL into Firefox: <http://<IP of Metasploitable2>/mutillidae>, and you will see the Mutillidae interface. If you see warnings from database, you should click the 'Reset DB' link in the Mutillidae interface to restore the database to its initial state. Then, the warnings should disappear. Note that, different from DVWA, you do not need to log into Mutillidae to access its pages. Also note that, the default security level of Mutillidae is '0' (the lowest security level) when you start browsing this application (see the screenshot below). This is the security level you should use during your pentest, and you should leave it as it is, i.e., never toggle it.




Mutillidae: Born to be Hacked

Version: 2.1.19
Security Level: 0 (Hosed)
Hints: Disabled (0 - I try harder)
Not Logged In

Home
Login/Register
Toggle Hints
Toggle Security
Reset DB
View Log
View Captured Data

Core Controls
OWASP Top 10
Others
Documentation
Resources



Site

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10


Latest Version / Installation

- [Latest Version](#)
- [Installation Instructions](#)
- [Usage Instructions](#)
- [Get rid of those pesky PHP errors](#)
- [Change Log](#)
- [Notes](#)

Mutillidae contains the pages corresponding to the OWASP Top 10 Security Risks. These pages can be accessed by the ‘OWASP Top 10’ menu located in the left. In this project, you are only required to pentest the SQLi page and the Stored XSS page among them. The detailed instructions are given below.

5.1 The SQLi page. [2 marks]

Click ‘OWASP Top 10’ → ‘A1 – Injection’ → ‘SQLi – Extract Data’ → ‘User Info’ as shown below.



Mutillidae: Born to be Hacked

Version: 2.1.19
Security Level: 0 (Hosed)
Hints: Disabled (0 - I try harder)
Not Logged In

Home
Login/Register
Toggle Hints
Toggle Security
Reset DB
View Log
View Captured Data

Core Controls
OWASP Top 10
Others
Documentation


A1 - Injection
A2 - Cross Site Scripting (XSS)
A3 - Broken Authentication and Session Management

SQLi - Extract Data
SQLi - Bypass Authentication
SQLi - Insert Injection
Blind SQLi via Timing

User Info

You will reach the ‘user-info.php’ page as shown below:

View your details


Back

Please enter username and password to view account details

Name

Password

View Account Details

Dont have an account? [Please register here](#)

In this page, enter crafted inputs for ‘Name’ and ‘Password’ respectively, such that the details of all users stored in the database are displayed. You should:

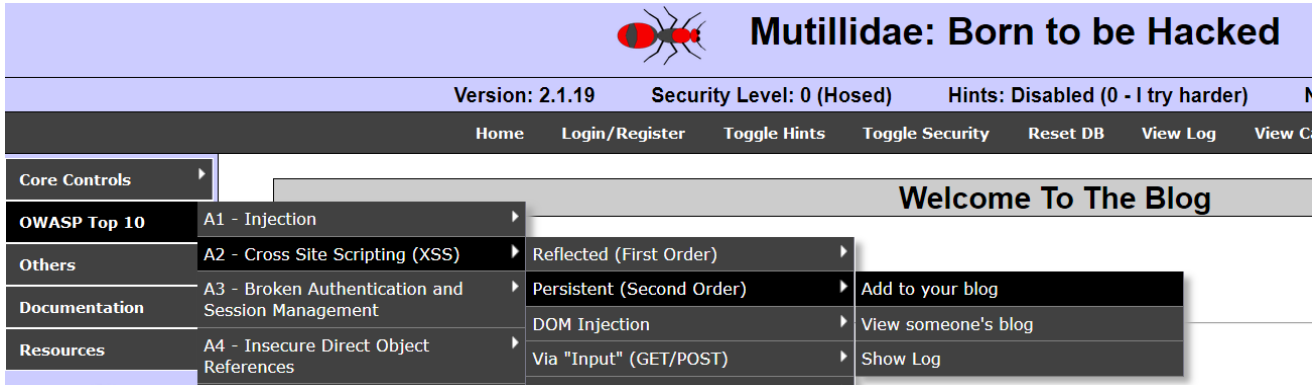
- Write your crafted inputs into the project report.
- Also, include a screenshot at least showing the details of the following two users: ‘admin’ and

'adrian'.

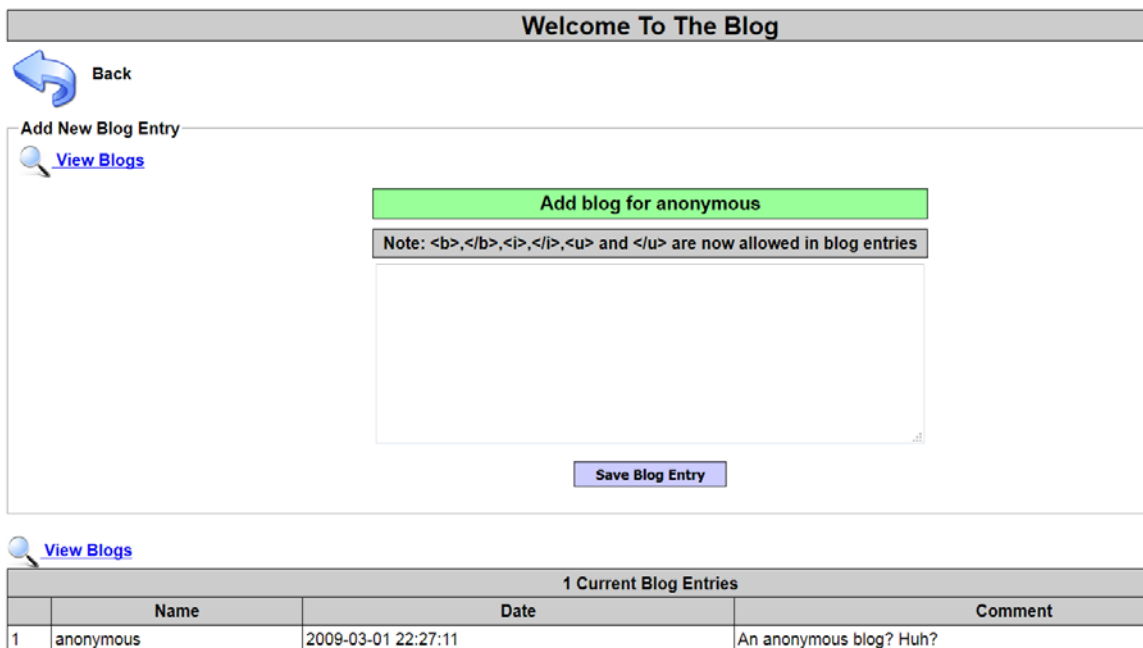
Hint: you can try the valid inputs (Name: admin, Password: adminpass) first.

5.2 The Stored XSS page. [3 marks]

Click 'OWASP Top 10' → 'A2 - Cross Site Scripting(XSS)' → 'Persistent(Second Order)' → 'Add to your blog':



You will reach the 'add-to-your-blog.php' page as shown below:




In this page, enter a crafted blog entry which can report the cookies of a web session to a server you set up. You should then use another browser to view this blog entry, and have the cookie for this new browsing session reported to the server you set up. This another browser can be the IE on WinXP VM. If you set up your own lab environment, you should make sure to have a third VM in it, such that you can use the browser on the third VM to browse your crafted blog entry.

You can follow Lecture 11 to achieve the above. In your project report, you should include the following:

- Detail your steps of setting things up such that when another browser visits your crafted blog entry, the cookie of this browsing session will be disclosed to you.
- Your crafted blog entry.
- A screenshot on the received cookies by the server you set up.

Marking Criteria

The mark allocation for each task is indicated at its end. We will conduct marking with a rubric reflecting this mark allocation. This rubric can be viewed by clicking the pentesting report submission link on vUWS, and then clicking the rubric icon  before the 'Submit' button.

In this rubric, there are three categories of marks for each task: Excellent, Adequate, and Incomplete. The meanings of these three categories are as follows:

- **Excellent (full mark):** Include all the command lines and critical screenshots such that another person can easily repeat what you have done, and both the steps and the results are correct. Moreover, use a narrative style similar to that in [the sample pentesting report from Offensive Security Ltd](#) to describe how you accomplish a task. We only require the narrative for each task, but not overall narratives such as executive summary. Your narrative should:
 - Has a label corresponding to its task label such as 1.1 or 3.1, etc.
 - Be easy to understand. If hard to understand, we will deem your steps incorrect.
 - Use full sentences.
 - Contain no more than 2 instances of the following: typos, grammatical errors, non-smooth sentences for each task.
- **Adequate (a mark a little less than full mark):** Include all the command lines and critical screenshots such that another person can easily repeat what you have done, and both the steps and the results are correct. However, the narrative fails to satisfy one of the requirements mentioned for 'Excellent'.
- **Incomplete (0 mark):** Any critical step or the result is missing or wrong. Note that we'll be strict with this one. So make sure to include all the command lines and critical screenshots for each task.

Report submission

- Attach the completed university assignment cover sheet to your report.
- Your report should be in PDF format. Name your report Surname-Givenname-StudentID.pdf.
- Submit to turnitin via the Project Submission link on vUWS website. Turnitin will calculate the similarity percentage of your report to other submissions. If you are detected with plagiarism by turnitin, you will be punished seriously according to university policy. Note: To prevent tricks against plagiarism detection, we will not show the similarity percentage to you after your submission.