

CSC427 - Metasploit Lab

Josh Alexander, Brandon Elefano

josh.alexander@mail.utoronto.ca, brandon.elefano@mail.utoronto.ca
alexa364, elefanob

University of Toronto Mississauga — March 07, 2022

1 Setup

1.1 Requirements

You will need to download the following programs/VMs:

- VirtualBox ([link](#))
- Our lab VM ([link](#))
- Kali Linux VirtualBox VM ([link](#))

1.2 Network Setup

After importing your VMs into VirtualBox, follow these instructions to setup your virtual network:

1. Go to the Host Network Manager (File » Host Network Manager)
2. Create a new network by clicking Create
3. Enable the DHCP Server on your newly created network by clicking on the corresponding checkbox

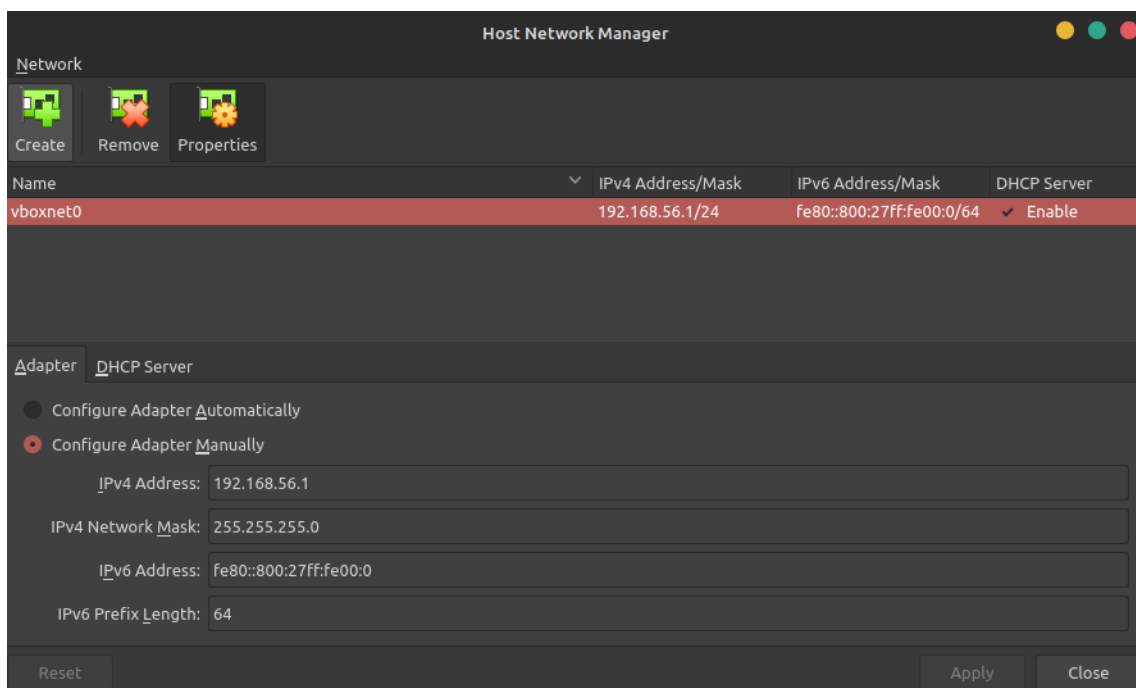


Figure 1: Host Network Manager

4. Exit out of the Network Manager, click on the lab VM, and go to Settings » Network
5. Make the following changes (if necessary)
 - Check the Enable Network Adapter box
 - Set Attached to: Host-only Adapter
 - Set Name: YOUR_NETWORK_FROM_STEP_2

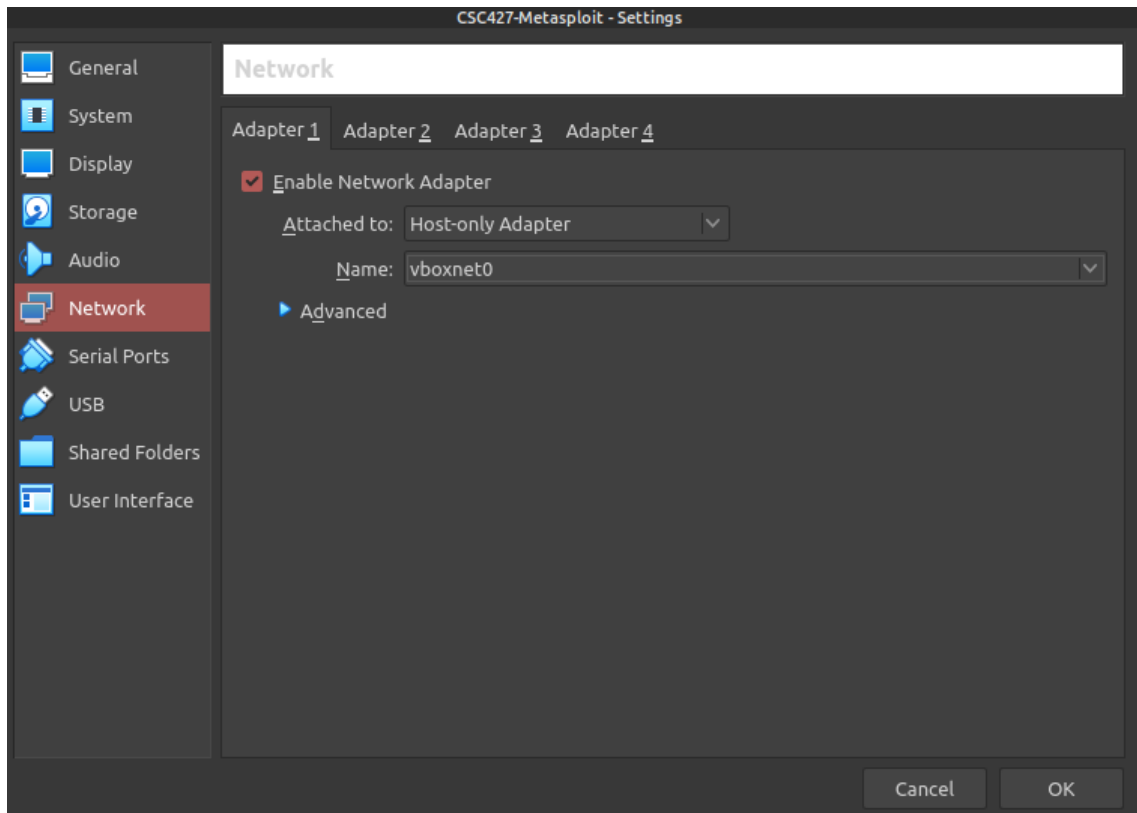


Figure 2: VM Network Settings

6. Perform steps 4-5 with the Kali VM

Congratulations! You are now ready to start the lab. Just start both VMs and log in to the Kali machine, and you can move onto the next section.

2 MSFconsole

In this section, we are going to gain root access into our lab VM using MSFconsole (Metasploit's CLI). Launch it in your Kali VM using the command `msfconsole`.

2.1 Enumeration

Right now, we don't know anything about our target except for the fact that it is connected to the same network our Kali machine is attached to. Let us see if we can learn more about it.

1. Using `db_nmap`, discover the IP address belonging to our Lab VM, figure out what ports are open, and determine the services running on each open port. You should be able to get all of this information using one command.

TODO: Submit the command you used to get this information.

2. Use the command `services` to get all of the service information gathered using `db_nmap`. How many ports are open on the target machine? What is the service running on the port with the lowest port number?

TODO: Submit your responses to these two questions.

2.2 Exploiting a Vulnerability

While it may seem that we learned little to nothing about our target system, we have actually stumbled on a critical piece of information! It turns out that there is a devastating vulnerability with the service running on the port with the lowest port number.

1. What is the CVE Number corresponding to this service?

TODO: Submit your response this question.

2. Use `search` to find a module that could help us exploit this vulnerability.
3. Notice that there is a number assigned to each module that matched our search query. We can type "`use 0`" to use the first module on the list, "`use 1`" to use the second module, etc. With this in mind, select a module that seems promising.
4. Type "`show options`" to show all of the parameters needed to run the module. Set the missing required parameters using the command "`set PARAMETER VALUE`" (where `PARAMETER`, `VALUE` are placeholders).
5. Execute the module using the command `exploit`. Now you have remote access to the target machine! What user are you logged in as?

TODO: Submit your response this question.

6. Try finding the flag hidden on the system. HINT: It is in someone's home directory.

TODO: Submit the hidden flag.

2.3 Meterpreter

While having a remote root shell is great, we can still do better. A meterpreter shell would post-exploitation tasks (exfiltrating information, leaving backdoors, etc.) a lot easier. Let us see if we can upgrade our remote shell.

1. Use `CTRL + Z` to background the session.
2. Search for the `POST` module `shell_to_meterpreter` and use it.
3. Set the missing parameters. You can get the session number of the shell you backgrounded in step 1 using the command `sessions`.
4. Execute the module using the command `exploit`. You can access the newly created meterpreter shell using the command `sessions SESSION_NO` (where `SESSION_NO` is the session number corresponding meterpreter session).
5. Play around with the shell. Type `help` to get helpful commands.
6. What is name of the target computer? HINT: What command gives you information about the remote system?

TODO: Submit your response this question.

7. What is the hash of the user "user"?

TODO: Submit the hash.

3 MSFvenom

In this section, we will be using `MSFvenom`, which is a tool that was designed to generate payloads.


3.1 Accessing DVWA

To start, we are going to visit the web application hosted by our lab VM.

1. Exit the previous session (CTRL + C). If we use the `services` command again, notice that port 80 is also open. Try accessing it using your browser.
2. Log into DVWA. The credentials are as follows:

Login: admin

Password: password



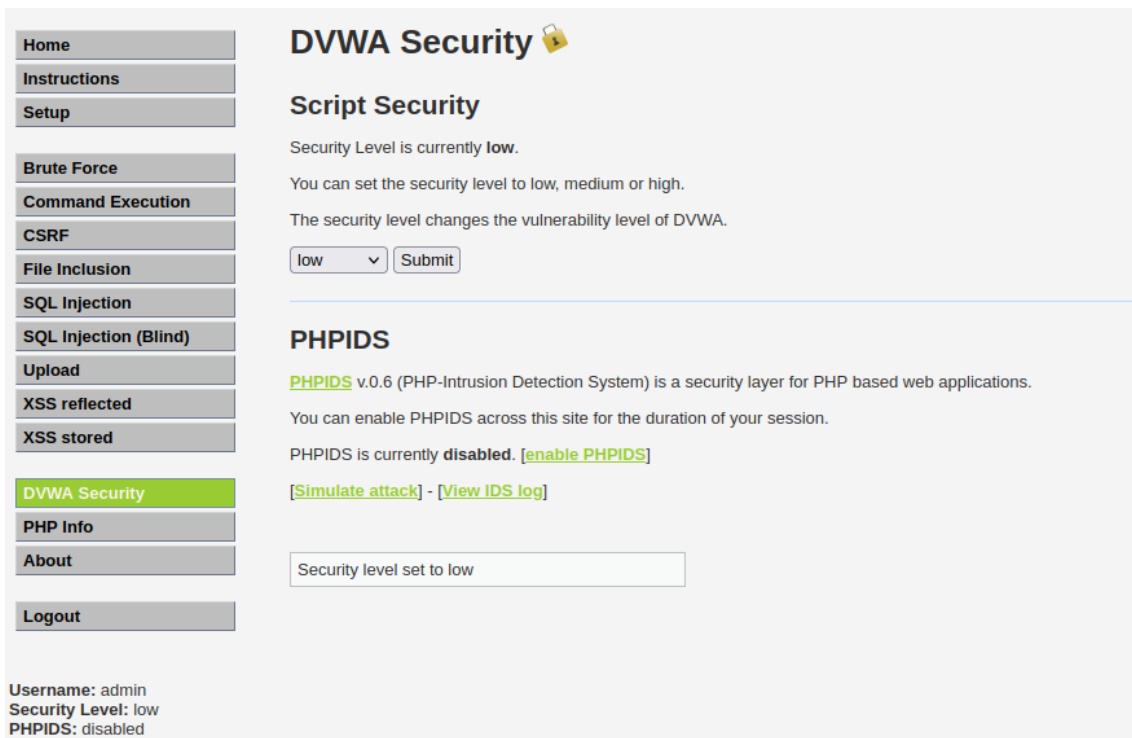
Username

Password

Login

Figure 3: DVWA Login Page

3. Go to DVWA Security and set the security level to low



The image shows the DVWA Security settings page. On the left is a sidebar with navigation links: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload, XSS reflected, XSS stored, DVWA Security (highlighted), PHP Info, About, and Logout. The main content area is titled 'DVWA Security' with a lock icon. Below the title is the 'Script Security' section, which states 'Security Level is currently low.' and 'You can set the security level to low, medium or high.' It also mentions 'The security level changes the vulnerability level of DVWA.' There is a dropdown menu set to 'low' and a 'Submit' button. Below this is the 'PHPIDS' section, which states 'PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.' It says 'You can enable PHPIDS across this site for the duration of your session.' and 'PHPIDS is currently disabled.' with links to '[enable PHPIDS]', '[Simulate attack]', and '[View IDS log]'. At the bottom, there is a box that says 'Security level set to low'. At the very bottom of the page, it displays 'Username: admin', 'Security Level: low', and 'PHPIDS: disabled'.

Figure 4: DVWA Security Settings

3.2 Building the Payload

Clearly, this application has a ton of vulnerabilities that are asking to be exploited. However, there are two "points of interest" that seem particularly promising: the *File Upload* page and the *Command Execution* page. It seems that we can use the latter page to execute any program uploaded using the former page. Let's try to build an executable that will exploit this.

1. Using `msfvenom`, generate an ELF file that will open a reverse shell to port 9001.

TODO: Submit the command you used to generate the payload.

3.3 Getting a Reverse Shell

Now that we have a payload, we can exploit the vulnerabilities mentioned in the previous section.

1. Start a netcat listener on port 9001.
2. Upload your payload to DVWA using the *File Upload* page. Take note of where uploads are stored.
3. Go to the *Command Execution* page and run the payload. HINT: If it isn't working, maybe you should check the file permissions.
4. What user is running the reverse shell?

TODO: Submit your response this question.

5. Prove that you got access to the system using photographic evidence.

TODO: Submit a screenshot showing that your netcat listener connected to the LabVM.

4 Submission

Please submit a PDF containing all of your answers to [Markus](#).