CS471 Assignment 6

Abstract

Design, implement, test, and demonstrate a basic network security ‘honeypot’. Analyze and discuss the results.

Assignment

Using a Kali Linux virtual machine and an Ubuntu Linux virtual machine, complete the following activity. Kali will be the probing or attacking system. Ubuntu will be the honeypot or target system. Setup the virtual machines to exist on the same network as the VM host by using ‘Bridged Adapter’ mode.

Provide a single document detailing the activity, including your process, methods, and results. All screenshots should be nicely resized and annotated. Your document should show what you did, how you did it, display the results, and explain what happened.

Include all of the files created during this activity with your submission. Additionally, include all of the commands used during your work in separate text file; this will also be included with your submission.

Activity

Simulate five legitimate network services
First, select 5 different network services to simulate as a honeypot. The Ubuntu system will host these services. For each service, create a listener that logs all connection attempts to a separate log file. Be sure each listener will continually log all connections to a file; it must not only log one connection. The listener must also reply back with something similar to the original service.

Next, run Wireshark to capture the traffic generated during the demonstration of the honeypot.

After all of the listeners are running, attempt to access the honeypot on the simulated services from the Kali VM. You may use the browser, nc, ssh client, nmap, or any other tool of your choice. Be sure to use at least 3 different tools to access simulated services.

An example of a listener service:
HTTP would be a good service to simulate. For this, listen on port 80 and respond with some text, “HTTP/1.1 200 OK\n\n”. This can be done with netcat. Run the following from the terminal in one copy/paste.

```
while true; do    echo -e "HTTP/1.1 200 OK\n\n $(date)" | nc -l -p 80 -q 1; done
```

To determine what each service replies when contacted, try connecting to a real server providing this service. Use netcat to connect, and notice the reply. Use this in your listener to simulate the service.

If testing SSH, be sure the SSH service is not running on the Ubuntu system before attempting to run a listener on the SSH service port, 22.

```
sudo systemctl stop ssh
```

Firewalls
It will be best to disable the Ubuntu firewall for this assignment.
```
sudo ufw disable
```

Accessing a listener service:

Attempt to access the listening service, running on Ubuntu, from the Kali VM. For each service, attempt to access it using the ‘appropriate’ client. For example, for a HTTP listener on port 80; this should be accessed using a web browser.

Ex. While a listener is running on port 80, open a browser to http://honeypot_ip_address:80.

Alternatively, you may want to use nmap to probe the open ports.
```
nmap -p 8000 -T4 -A -v honeypot_ip_address
nmap -p 1-65535 -T4 -A -v honeypot_ip_address
```

Analysis

Review the honeypot logs and results. Present your results in an informative way.

Review all of the packets captured by Wireshark. Present your results in an informative way.

Filter and save only several relevant packets. Include this filtered capture file.

Conclusion

In addition to your conclusion, add the following:
- How can a honeypot be used to improve security?
- What kind of information could be gathered about the techniques used by attackers?
- Honeypots are passive. How could this be combined with a firewall to create a more active solution?

Deliverables

A document detailing the activity, including your process, methods, and results. This includes annotated screenshots. Clearly detail your work in a reproducible way following the provided sample format. Do not provide any image or text from another source without citation.

Submit all files created for this assignment. Attach these files to your submission. Do not zip, tar, or archive. The packet capture files should only include the requested files; these should be fairly small files.

Additionally, submit a single text file with all of the commands used for this assignment. One command per line. This should be complete and organized in order of use.

Upload these files to Canvas before the deadline.