Project 3: Network Security

This project is due on **Wednesday, March 8** at **6 p.m.** and counts for 8% of your course grade. Late submissions will be penalized by 10% plus an additional 10% every 5 hours until received. Late work will not be accepted after 19.5 hours past the deadline. If you have a conflict due to travel, interviews, etc., please plan accordingly and turn in your project early.

This is a group project; you will work in **teams of two** and submit one project per team. Please find a partner as soon as possible. If you have trouble forming a team, post to Piazza's partner search forum. The final will cover project material, so you and your partner should collaborate on each part.

The code and other answers your group submits must be entirely your own work, and you are bound by the Honor Code. You may consult with other students about the conceptualization of the project and the meaning of the questions, but you may not look at any part of someone else's solution or collaborate with anyone outside your group. You may consult published references, provided that you appropriately cite them (e.g., with program comments), as you would in an academic paper.

Solutions must be submitted electronically via Canvas, following the submission checklist below.

Introduction

This project will introduce you to common network protocols, to network packet trace analysis, and to the basics of network penetration testing.

Objectives

- Gain exposure to core network protocols and concepts.
- Learn to apply manual and automated traffic analysis to detect security problems.
- Understand offensive techniques used to attack local network traffic.
- Practice network penetration testing.

Read this First

This project asks you to perform attacks, with our permission, against a target network that we are providing for this purpose. Attempting the same kinds of attacks against other networks without authorization is prohibited by law and university policies and may result in *fines, expulsion, and jail time*. **You must not attack any network without authorization!** Per course policy, you are required to respect the privacy and property rights of others at all times, *or else you will fail the course.* See "Ethics, Law, and University Policies" on the course website.

Part 1. Exploring Network Traces

Security analysts and attackers both frequently study network traffic to search for vulnerabilities and to characterize network behavior. In this section, you will examine a network packet trace (commonly called a "pcap") that we recorded on a sample network we set up for this assignment. You will search for specific vulnerable behaviors and extract relevant details using the Wireshark network analyzer, which is available at https://www.wireshark.org.

Download the pcap from https://eecs388.org/*/388-proj3.pcap, and examine it using Wireshark. Familiarize yourself with Wireshark's features. and try exploring the various options for filtering and for reconstructing data streams.

Concisely answer the questions below. Each response should require at most 2–3 sentences.

1. Multiple devices are connected to the local network. What are their MAC and IP addresses?

2. What type of network does this appear to be (e.g., a large corporation, an ISP backbone, etc.)? Point to evidence from the trace that supports this.

3. One of the clients connects to an FTP server during the trace.

   (a) What is the DNS hostname of the server it connects to?

   (b) Is the connection using Active or Passive FTP?

   (c) Based on the packet capture, what's one major vulnerability of the FTP protocol?

   (d) Name at least two network protocols that can be used in place of FTP to provide secure file transfer.

4. The trace shows that at least one of the clients makes HTTPS connections to sites other than Facebook. Pick one of these connections and answer the following:

   (a) What is the domain name of the site the client is connecting to?

   (b) Is there any way the HTTPS server can protect against the leak of information in (a)?

   (c) During the TLS handshake, the client provides a list of supported cipher suites. List the cipher suites and name the crypto algorithms used for each.

   (d) Are any of these cipher suites worrisome from a security or privacy perspective? Why?

   (e) What cipher suite does the server choose for the connection?

5. One of the clients makes a number of requests to Facebook.

   (a) Even though logins are processed over HTTPS, what is insecure about the way the browser is authenticated to Facebook?

   (b) How would this let an attacker impersonate the user on Facebook?

   (c) How can users protect themselves against this type of attack?

   (d) What did the user do while on the Facebook site?

What to submit Submit a text file containing your answers. Make sure each answer is formatted as a single line. Format your file using this template:

```
# Question 1

1. [Answer ...]

# Question 2

2. [Answer ...]

# Question 3

3a. [Answer ...]

3b. [Answer ...]

3c. [Answer ...]

3d. [Answer ...]

# Question 4

4a. [Answer ...]

4b. [Answer ...]

4c. [Answer ...]

4d. [Answer ...]

4e. [Answer ...]

# Question 5

5a. [Answer ...]

5b. [Answer ...]

5c. [Answer ...]

5d. [Answer ...]
```

Part 2. Anomaly Detection

In Part 1, you manually explored a network trace. Now, you will programmatically analyze a pcap file to detect suspicious behavior. Specifically, you will be attempting to identify port scanning.

Port scanning is a technique used to find network hosts that have services listening on one or more target ports. It can be used offensively to locate vulnerable systems in preparation for an attack, or defensively for research or network administration. In one kind of port scan technique, known as a `SYN` scan, the scanner sends TCP `SYN` packets (the first packet in the TCP handshake) and watches for hosts that respond with `SYN+ACK` packets (the second handshake step).

Since most hosts are not prepared to receive connections on any given port, typically, during a port scan, a much smaller number of hosts will respond with `SYN+ACK` packets than originally received `SYN` packets. By observing this effect in a packet trace, you can identify source addresses that may be attempting a port scan.

Your task is to develop a Python program that analyzes a pcap file in order to detect possible `SYN` scans. To do this, you will use `dpkt`, a library for packet manipulation and dissection. It is available in most package repositories. You can find more information about `dpkt` at https://github.com/kbandla/dpkt and view documentation by running `pydoc dpkt`, `pydoc dpkt.ip`, etc.; there's also a helpful tutorial here: https://jon.oberheide.org/blog/2008/10/15/dpkt-tutorial-2-parsing-a-pcap-file/.

Your program will take the path of the pcap file to be analyzed as a command-line parameter, e.g.:

```
python2.7 detector.py capture.pcap
```

The output should be the set of IP addresses (one per line) that sent more than 3 times as many `SYN` packets as the number of `SYN+ACK` packets they received. Your program should silently ignore packets that are malformed or that are not using Ethernet, IP, and TCP.

A large sample pcap file captured from a real network can be downloaded at ftp://ftp.bro-ids.org/enterprise-traces/hdr-traces05/lbl-internal.20041004-1305.port002.dump.anon. (You can examine the packets manually by opening this file in Wireshark.) For this input, your program's output should be these lines, in any order:

```
128.3.23.2
128.3.23.5
128.3.23.117
128.3.23.158
128.3.164.248
128.3.164.249
```

What to submit Submit a Python program that accomplishes the task specified above, as a file named `detector.py`. You should assume that `dpkt` 1.8 and `scapy` 2.2 are available, and you may use standard Python system libraries, but your program should otherwise be self-contained. We will grade your detector using a variety of different pcap files.

Part 3. Penetration Testing

The fictional company SuperDuperSketchyCorp has contracted with EECS 388 to provide penetration testing services to it in exchange for free hugs and awesome memes. Each project team will conduct a thorough penetration test of the company's networks and exposed systems.

Before you begin This part of the project spec serves as a Pen Test Engagement Agreement, covering the goals, scope, compensation, and authorization to begin the penetration test. You must agree to these terms in writing (as explained below) before you begin your work.

Contact information General questions should be posted to Piazza. We encourage giving each other help, but do not post spoilers (hints that give away the "Aha!" moments) or detailed instructions. Questions about potential rule-breaking should be emailed to eecs388-proj3@umich.edu.

Introduction

SuperDuperSketchyCorp recently set up a remote office in the Bob and Betty Beyster Building (BBB) for its employees to work in. SuperDuperSketchyCorp is concerned that its remote office may be more vulnerable than its headquarters since it uses a wireless network to provide access to its remote employees.

SuperDuperSketchyCorp has shared a diagram of its infrastructure, as shown in Figure 1.

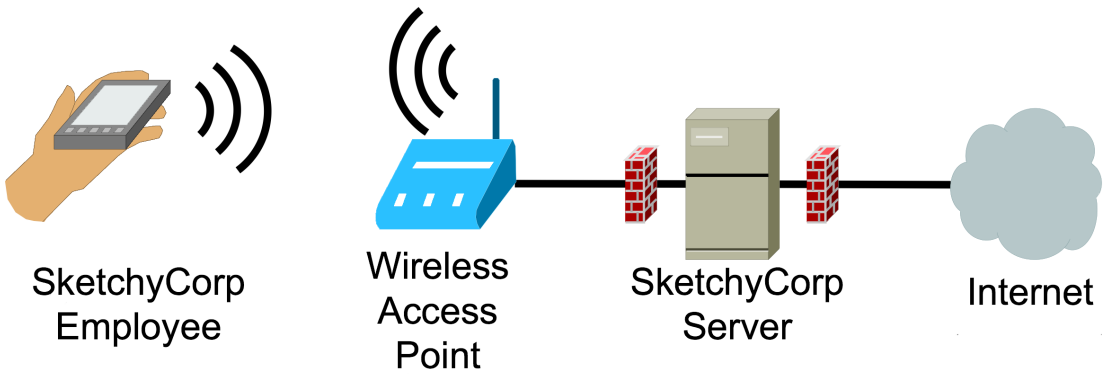

Figure 1: Infrastructure overview of SuperDuperSketchyCorp's remote office.

SuperDuperSketchyCorp employees connect to the **SuperDuperSketchyCorp_388** wireless network using WPA2-PSK security settings. From there, they can access the SuperDuperSketchyCorp server, which allows company employees to log in and gain access to company proprietary information. It just so happens that there is an employee at SuperDuperSketchyCorp with your uniqname.

Your objective is to test the security of SuperDuperSketchyCorp's networks and systems. In this engagement you will be authorized to break in to SuperDuperSketchyCorp's systems and explore any vulnerabilities you find, subject to the Rules of Engagement below. As in a real-world penetration test, you will be expected to use your ingenuity and technical skills to discover clues and techniques for meeting your objectives.

Deliverables

This contract stipulates that by the project due date, your team will submit to Canvas a test report written for a technical audience containing the following sections:[1]

Overview	A 2–3 sentence description of the objective of the pen test.
Methodology	A 1–2 paragraph description of the work you performed.
Findings	A description of any findings you may have made. Specifically, include details about the following: • Hostnames of any machines you gain access to during the pen test. • Any encryption keys for networks you gain access to. • Any credentials you are able to obtain (not including your own). • Other company secrets you accessed (briefly list each one). • [**Extra credit**] What is Bob hiding? (If you encounter a file or folder with "bonus" in its name, it pertains entirely to this optional question.)
Remediation	A 1–2 paragraph recommendation for what SuperDuperSketchyCorp should do to secure its remote office. You should address each of your findings.

Rules of Engagement

There will be certain systems and networks that are *in scope* for this project. Everything else should be considered *out of scope*. If you have any questions about what is in or out of scope for this project, get clarification from one of the course TAs *before* you act.

Things that are in scope: (*Hint: These are all things you should be doing.*)

- Capturing network traffic on the SuperDuperSketchyCorp wired and wireless networks.
  [Note: `enp2s0` on the SuperDuperSketchyCorp server is a U-M network and out of scope. `wlx00c0ca852fce` is a SuperDuperSketchyCorp network and in scope. Make sure you specify this correctly, since the default is to capture on `enp2s0`.]
- Using automated network scanning tools to break into the SuperDuperSketchyCorp network.
- Connecting to the SuperDuperSketchyCorp wireless network.
- Logging in to SuperDuperSketchyCorp systems with employee credentials you obtain.
- Sending phishing emails to `helpdesk@email.superdupersketchycorp.com`.

Everything not explicitly in-scope is out of scope, and should not be attempted.

[1] This test report structure is loosely based on the SANS outline: https://www.sans.org/reading-room/whitepapers/bestprac/writing-penetration-testing-report-33343.

Here are a few examples of activities that are **out of scope** for this project:

- DO NOT scan the SuperDuperSketchyCorp server from MWireless or MGuest. (From the project-specific wireless network, it's fine.)
- DO NOT perform actions that cause difficulty for other users or that interfere with the project infrastructure (i.e., denial of service).
- DO NOT attempt to elevate your shell privileges on the SuperDuperSketchyCorp server. Use it as you would the CAEN computers.
- DO NOT do anything else that's not specifically designated as in scope. If you're unsure, please ask for clarification.

A note about cheating: There may be backdoors you discover along the way. If these are shared with you by someone else before you discover them yourself, DO NOT USE THEM. We will be auditing the progress of each team, and, if we see you skip steps, we will consider this cheating. After you discover a backdoor on your own, you may use it. If you have questions about whether you may use a particular backdoor, post a private question on Piazza *before* using it.

Compensation

This part is worth 3% of your course grade, of the 8% this project is worth, and will itself be graded out of 80 points. It will be graded as follows:

- Test report with all sections as described:
  Overview [4 pts.]
  Methodology [11 pts.]
  Findings, not including points for specific findings [3 pts.]
  Remediation [14 pts.]
- Notable findings as described in the findings section [each 8 pts., max 48 pts.]
  (Username and password pairs count as one finding, as do port and hostname pairs, and any collection of secrets found in one file or web page.)
- The optional extra credit: What is Bob's secret? [10 bonus pts.]

Authorization

This document authorizes you, subject to the terms and conditions herein, to begin the penetration test for SuperDuperSketchyCorp on behalf of EECS 388. To accept this agreement and begin, you must email your acceptance ("I accept the EECS 388 Pen Testing Agreement") to *eecs388-proj3@umich.edu* along with the maximum number of years in **jail** that you could face under 18 USC § 2511 for intercepting traffic on an encrypted WiFi network without permission.

Both team members **must** send this email before you begin.

Click here for a mailto: link with prepopulated fields.

Important Tips

Attend lab for important setup instructions and introductions to several networking tools and techniques. You will need these to complete the project.

Once you reach the SuperDuperSketchyCorp server, you will discover you no longer need to use the SuperDuperSketchyCorp wireless network at all. Please consider disconnecting from it to reduce wireless congestion for your classmates.

Please do not kick anyone off WiFi for this project. We have pre-recorded capture files for the SuperDuperSketchyCorp network, which you can find here: https://eecs388.org/*/388-proj3-wifi.zip.

Figure 2: A flier that was found in the BBB lobby. Looks important.

Tool Box

Here is a partial list of tools that may be helpful for this project. This list is not complete. You will need to use tools beyond the ones listed here.

- **Kali Linux**. Linux distribution used for penetration testing, ethical hacking, and network security assessments. https://www.kali.org/downloads/

  See also: How to enable the network in Kali Linux Virtual Box.

- **man**: The manual. Use this command to read the manual for other commands, including their options. Where man is unclear, Google will be extremely helpful for this assignment.

- **aircrack-ng**: Cracks wireless passwords. Generally requires a network traffic capture. See also: How To Hack WPA/WPA2 Wi-Fi With Kali Linux and Aircrack-ng (start at step 11).

- **nmap**: Network exploration tool and port scanner. Can scan network to find hosts, find open ports, even detect software versions in come cases.

- **tcpdump**: Network traffic analysis tool. Can capture traffic and save to a file. Can view traffic in real time. The `-A` and `-w` options may be helpful for this project.

- **Wireshark**: Graphical network traffic analysis tool. Network traffic captured with tcpdump can be viewed with Wireshark.

- **ssh**: Login to servers remotely.

- **scp**: Secure copy. Uses the ssh protocol to transfer files between hosts.

- **nc**: netcat. Can do almost anything TCP or UDP related. Read about how it can be used to spoof email.

Submission Checklist

Upload to Canvas a gzipped tar file (`.tar.gz`) named `project3.`*`uniqname1`*`.`*`uniqname2`*`.tar.gz` that contains only the files listed below. Make sure you have the proper filenames and behaviors. You can generate the tarball at the shell using this command:

```
tar -zcf project3.uniqname1.uniqname2.tar.gz \
  pcap.txt detector.py report.txt
```

Part 1: Exploring Network Traces

`pcap.txt` A text file containing your answers to the questions in Part 1.

Part 2: Anomaly Detection

`detector.py` Your plain text Python program for `SYN` scan detection.

Part 3: Penetration Test

`report.txt` A text file with the contents specified in Part 3.