Homework 4: Denial of Service

This homework is due **Wednesday, March 15** by **6 p.m.** and counts for 5% 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 homework early.

We encourage you to discuss the problems and your general approach with other students in the class. However, the answers you turn in must be your own original work, and you are bound by the Honor Code. Solutions must be submitted electronically via Gradescope in PDF format by completing the template at the end of this document.

Concisely answer the following questions. (Limit yourself to at most 80 words per subquestion.)

1. **Client puzzles and amplification** Denial-of-service (DoS) attacks attempt to overwhelm a server with a huge volume of requests. Researchers have proposed a defense against DoS attacks called *client puzzles*: For each request, the server sends the client a freshly generated random challenge r and a difficulty parameter n, and the client has to produce a solution s such that the last n bits of $\text{HMAC}_r(s)$ are all 0. Clients must present a valid solution to receive service.

   (a) What is the expected number of HMAC computations for the client to compute the solution? How many HMAC computations does it take for the server to check the solution?

   (b) Suppose a "unit of work" is equivalent to the difficulty of computing one HMAC. If an attacker enjoys an *amplification factor* of 64 (i.e., the attacker can cause the server to do 64 units of work by expending one unit of work), what should n be to negate this advantage using client puzzles?

   (c) Some denial-of-service attacks attempt to exhaust the victim's network bandwidth rather that its CPU resources, and amplification plays an important role in such attacks too. Attackers recently began exploiting the Network Time Protocol (NTP) for this purpose, as described in this article: https://goo.gl/oVXH7V. What two features of NTP make it an ideal DoS tool?

2. **The Mirai IoT botnet** US-CERT defines the Internet of Things (IoT) as an "emerging network of devices (e.g., printers, routers, video cameras, smart TVs) that connect to one another via the Internet, often automatically sending and receiving data." Although IoT devices typically have limited bandwidth and processor power, on the Internet today, they represent a far greater share of public IP addresses than traditional servers.

   Attackers have started to enlist IoT devices to create large-scale botnets. In September 2016, security blogger Brian Krebs was targeted by one of the largest DDoS attacks ever observed, with more than 620 Gbps of traffic directed at his website by an IoT botnet called Mirai. You can read more about the attack in this US-CERT bulletin: https://goo.gl/H5UbYV. In case you're curious, the source code for the Mirai bot is here: https://goo.gl/z05omT.
   We do not recommend installing or using Mirai!

   (a) Based on the US-CERT bulletin above, how much bandwidth did each Mirai bot contribute to the attack on Brian Krebs, on average? How does this compare to the bandwidth of a typical home Internet connection in the U.S.?

   (b) Briefly explain what strategy *Mirai* uses to discover and infect IoT devices.

   (c) Based on this strategy, what could the owner of the device do in order to prevent a successful hack and how will this change help?

   One of the preventative steps suggested by US-CERT is to "disable Universal Plug and Play (UPnP) on routers unless absolutely necessary."

   (d) Briefly, explain what UPnP is and what it is used for.

   (e) What is the problem with UPnP, and how does disabling it help prevent attacks on your devices? (This paper by Armijn Hemel might be helpful: https://goo.gl/n3R6a.)

3. **Distributed denial-of-service** A popular attack tool among novice hackers previously has been the Low Orbit Ion Cannon (LOIC), which features a user-friendly GUI as well as an option to voluntarily add yourself to a botnet controlled via an IRC channel.
   We do not recommend installing or using LOIC!

   (a) LOIC is a fairly simple program. The source file at https://goo.gl/6d4EUX contains the primary attack mechanism. Briefly, how does this mechanism work?

   (b) The LOIC command and control system ("Hive Mind mode") is also fairly simple. It is described in the README file at https://goo.gl/mpu9ZU. Briefly, how does this mechanism work?

   (c) Other than client puzzles, what are some things a website could do to defend itself against a LOIC Hive Mind attack? If the attack involves thousands of bots, how can the server distinguish them from legitimate clients?

   (d) Briefly, compare and contrast LOIC Hive Mind mode to a typical botnet.

Submission Template

Please submit one file to Gradescope: **submission.pdf**. Make sure each answer is formatted as a single line, and that the file you submit is in `.pdf` format.

```
# Problem 1

1a. [Answer ...]

1b. [Answer ...]

1c. [Answer ...]

# Problem 2

2a. [Answer ...]

2b. [Answer ...]

2c. [Answer ...]

2d. [Answer ...]

2e. [Answer ...]

# Problem 3

3a. [Answer ...]

3b. [Answer ...]

3c. [Answer ...]

3d. [Answer ...]
```