Yu-Tung Pai (yp332)
Maureen Hsu (yh863)

Short Answer Questions 1.5:

- Why does using GCM prevent mauling and padding oracle attacks?

CBC leaks partial information about the result of decryption. Specifically, in CBC, previous block hashing will be used in the next block. GCM, on the other hand, does not have this dependency. It uses a counter for each block of data, then uses that number to create the ciphertext. Therefore, it is not susceptible to padding oracle attacks.

- For each attack above, explain whether enabling HTTPS for the entire payment site (as opposed to just the login page) prevents the attack if no other countermeasure is applied.

Yes, HTTPS can help both. For mauling, in this way attacks wouldn't know the cookie content. Thus, they wouldn't know the concatenation rule and be able to manual cookies. For padding oracle, they also can't get the content under the protection of HTTPS.

Short Answer Questions 2.3:
This attack relies on the attacker having knowledge of the SipHash key. Assuming SipHash is a pseudo-random function, does sampling a fresh random SipHash key for the hash table every time the web server is started prevent this attack? Why or why not?

Yes, the attacker will have to find the key first each time the web server restarts. The attacker will have to pay more effort on attacking and thus it prevents the attacker from this attack.

Short Answer Question 2.5: One suggested countermeasure to denial-of-service attacks is proof of work: forcing clients to perform some computational work and including a proof in the request. Is this an effective countermeasure? Why or why not?

Yes and no. In some circumstances, the expensive computational cost might prevent the attacker from attacking. However, if the attacker has very large computational power (larger than the requested as proof of work) or the attacker can infect other users and make the request sent from their computer, the attacker does not even need to pay any additional computational effort from the attacker's side.