CMPSC 177 Homework 2

Task 1

1. Since there are only two messages that Mr. Cipher uses, and one message is clearly shorter than the other, all we have to do is count the number of blocks in the encrypted message. If the number of blocks is the same as it is on most days, than it is the "Nothing to report" message. Otherwise, if it's longer than normal, than Mr. Cipher will be at the rendezvous point the next day.
2. Mr. Cipher can pad both of his messages so that they both output the same number of encrypted blocks when put through the encryption function. This way, determining which message is which is not as simple as checking the number of blocks in the message.

Task 2

1. If *H* follows the MD paradigm, then *H* does not satisfy the unforgeability of a MAC, as it is vulnerable to a length extension attack. This is possible even without knowing the secret key K as the input message M, which is known, can be used to adjust the internal states of the function. Then, that internal state can be used to generate a valid signature for a different message M', which the server will recognize as a valid request.
2. In an HMAC, this kind of an attack is not possible as it is a secret-key primitive, where the only way to evaluate the message correctly is by having the key, unlike with an MD hash function which can be fooled even without the key, as shown above.

Task 3

1. I can infer that the plaintexts of the two ciphertexts are the same. This is because the authentication code at the end (starting with 7eb and ending with 5e9) are the exact same. This is because encrypt-and-MAC presents the same message to both the encryption and the MAC function, and while encryption functions can output different ciphertexts for the same message due to cipher-block-chaining, MACs must provide the exact same tag for the same message.
2. E&M can never be fully semantically secure, as the MAC tag at the end can hint at the contents of the original plaintexts. If the authentication codes are the same, the plaintexts are the same. While that information on its own may not be enough to damage either integrity or confidentiality, an authenticated encryption scheme should reveal nothing about its contents at all.
3. Yes, E&M does reasonably satisfy integrity, as it's not possible to fool the authentication code without the key, and while the encrypted part of the ciphertext might be changed, the authentication code at the end cannot be spoofed.

Task 4

1. Regardless of whether or not the plaintext was correctly padded, the decryption function will assume that it was correctly padded. Thus, the attack that we make has to provide valid padding. This is guaranteed if eitheror xored to the last byte of the ciphertext results in the last byte being 0x01 or the last byte being the same as the second-to-last byte, whatever that may be. Both will result in correct decryption, but the former will take more time than the latter.
2. The validity check that we do is to change the second-to-last byte. The guess that remains valid after this change is the correct value.
3. We extend the padding oracle attack moving on to the next byte to the left and xoring by the number of bytes we’re trying to cover. For example, we xor 0x02 while trying to figure out the second-to-last byte, and by 0x03 for the third-to-last, and so forth. And each time, we do validity checks except for the first block, which means we have 15 validity checks for a single 16 byte block.