# R9 In what way does a hash provide a better message integrity check than a checksum (such as the Internet checksum)?

A (good) hash function provides unique finger print for a message/file, while a checksum only has rudimentary error detection capabilities, it is easy to construct a new message with the same checksum, thereby the checksum us not good for integrity.

# R10. Can you “decrypt” a hash of a message to get the original message? Explain your answer.

Multiple messages can hash to the same value. Therefore there would be ambiguity when trying to “decrypt” a hash, and it is therefore not possible. (Hashes have been broken where a large table of different hashes have been created, therefore given a hash there is a good chance you know what message what used to generate it).

# R16. What is the purpose of a nonce in an end-point authentication protocol?

Making sure each session is encrypted with unique session key. Thereby defending against repeated attacks.

# P15. Consider our authentication protocol in Figure 8.18 in which Alice authenticates herself to Bob, which we saw works well (i.e., we found no flaws in it). Now suppose that while Alice is authenticating herself to Bob, Bob must authenticate himself to Alice. Give a scenario by which Trudy, pretending to be Alice, can now authenticate herself to Bob as Alice. (Hint: Consider that the sequence of operations of the protocol, one with Trudy initiating and one with Bob initiating, can be arbitrarily interleaved. Pay particular attention to the fact that both Bob and Alice will use a nonce, and that if care is not taken, the same nonce can be used maliciously.)

A white board with writing on it

Description automatically generated

Bob tries to authenticate himself. By sending R, and receiving R. And then send KA-B(R), and receive KA-B(R), Bob then expects the other in the communication is Alice as she is the only one with KA-B.

But Trudy can intercept the message and just reply with the same values as, Bob is sending her.

This is useful as Trudy can log all encrypted messages between A and B. And the once (enough years later) she is capable of breaking the key she can read the messages.

# P18. Suppose Alice wants to send an e-mail to Bob. Bob has a public-private key pair (KB +, KB -), and Alice has Bob’s certificate. But Alice does not have a public, private key pair. Alice and Bob (and the entire world) share the same hash function H(# ).

## In this situation, is it possible to design a scheme so that Bob can verify that Alice created the message? If so, show how with a block diagram for Alice and Bob.

No, Alice have no information to differentiate her from anyone else.

## Is it possible to design a scheme that provides confidentiality for sending the message from Alice to Bob? If so, show how with a block diagram for Alice and Bob.

Alice encrypts message using Bob public key and sends it, Bob is the only with the private key capable of decrypting the message. (Thus, confidentiality is preserved)

Text

Description automatically generated

The text says that Bob’s key is unique to him, which implies that nobody else should know it.

The goal is for Alice to verify that the message was from Bob. But in order for her to do this she needs to compute the hash of the MAC which