Laboratory exercise 2, Computer Security

This exercise is a natural follow-up of Lab 1. For all instructions, please refer to Lab 1. You should not start with this lab until you have successfully finished Lab1.

In this lab, you are required to finish the following tasks:

Task 3: Verify the Message Authentication Code

Now that you have decrypted the ciphertext and it is time to calculate the message authentication code (MAC). Use key2 and your decrypted plaintext to calculate a mac. Compare your calculated mac with the two mac files (in hex string) in the .zip file. Which mac is the correct one?

Task 4: Verify the Digital Signature

As a receiver, to verify the digital signature we need to read the sender's public key that is currently stored in the public key certificate file lab1Sign.cert. You need to read the file and then load the public key with the java class CertificateFactory, or with the command window tool 'keytool' (not recommended). When you have the public key, you can check which of the two given signature files is correct.
The digital signature has been signed using the SHA1withRSA algorithm, and used the plaintext and the sender's private key as inputs.

Open question:

Suppose the receiver (i.e. you) does not share any secret with the sender before she/he receives the encrypted keys in ciphertext.enc (i.e. the ciphertext + the encrypted symmetric keys). Does a verified correct message authentication code (MAC) (e.g. the one received by applying HmacMD5 in this exercise) authenticate the sender or can we trust the origin of the message in this case? Why or why not? (Note that we are assuming that digital signature is not used)