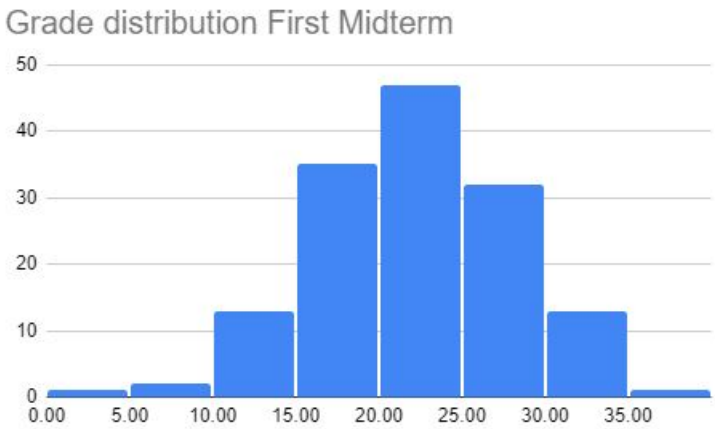
# COM-301 - First Midterm
# Most Repeated Errors

November 17, 2019




## General Advice

- When you are asked to write a response in three lines, respond in three lines. it is ok to be a bit wider or have couple of words in the next line due to some correction. We did not consider answers that blatantly ignored the instructions.
- When you are asked to choose a property, an approach, identify one technology, provide more than just the name. The goal of the question is to check that you understand these concepts and how to apply them. Providing only the name does not give points (unless the question is: what is the name of...?)

### Multi-choice Question: Authentication

*Error: Salts increase pre-image resistance.* Recall that salting the passwords means to store them as follows:

`H(password||salt), salt.`

The role of the salt is to avoid mass offline attacks in which the adversary can pre-compute the hash of all passwords, H(password), by forcing them to compute H(password——salt) for all possible salts (unfeasible if the salt is long enough). The salt itself does not strengthen pre-image resistance of the hash function. The hash function is the same regardless of the salt.

**Multi-choice Question: Cryptography**

*Error: The primitive required is a hash function with collision resistance.* The participants create a hash, such that if this is intercepted, it is difficult to find the message that corresponds to this hash. This is the pre-image resistance property, not the collision resistance.

**Multi-choice Question: Security policies**

*Error: Chinese policy forbids to work with clients whom you worked with in the past.* The goal of the Chinese Wall policy is to avoid information flows between entities competing with each other. Thus, if an agent works for Coop cannot later work for Migros, but naturally can work for Coop again. There is no risk of unauthorized information flow.

**Short Question: Security principles**

*Error: Firewalls do not fulfill least privilege.* A firewall decision is binary: either it lets the flow in, or it stops the flow. Every flow that is allowed into the network has the same privilege: going into the network. Therefore, there is no "least privilege" (you can also see it as all flows are allowed in with the least privilege).

**Short Question: Cipher configuration**

*Error: A cipher is insecure if the IV is repeated.* Remember from the lesson Applied Cryptography that the rule is "No IV reuse **under the same key**" (slide 18). What has to be unique is the tuple (IV, key). The function in the exam contained the instruction:
`key = GenerateSecureRandomKey() // generates a truly random key`
Therefore, even thought the IV does not change, and it is public, the configuration is secure.

**Short Question: Cryptography**

*Error: Adding a MAC always guarantees integrity.* Cryptographic primitives reduce security of the content to the security of the key. In the case of MAC algorithms, which provide authentication and integrity in a symmetric setting, these properties depend on the confidentiality and integrity of the shared symmetric key. In the exam, Alice created a fresh symmetric key and send it to Bob encrypted with Bob's public key. This guarantees key confidentiality, but *Error: does not guarantee key integrity.* An adversary can intercept the message, create his own symmetric key, encrypt it with Bob's public key, encrypt any message with the symmetric key, and compute a correct MAC. When receiving this message, Bob has no means to determine if the content is the original sent

by Alice, or the one created by the adversary.

**Short Question: Security Principles**

*Error: The mechanism is not open design.* The design is open! The only secret is what port has been selected every time the service is launched.

**Short Question: Authentication**

*Error: Encryption prevents replay attacks.* In replay attacks the adversary can observe and record the "message" that a user sends to another entity to authenticate. Then, the adversary can replay this message to authenticate as the user. (Lecture authentication, slides 10-11). Note that in these slides, the message is encrypted! What prevents replay attacks is freshness, either by using a fresh secret every time, or by using a fresh challenge used to prove that the reply has not been generated in the past.
*Error: Note:* We are also not very clear as to how one can perform Diffie-Hellman, RSA, or encryption on knocks on a door...

**Short Question: UNIX permissions**

*Error: Need for sticky or suid bits.* Not all problems require the use of sticky or suid bit. TAs have modify access on the binary file and setting suid on the file is equivalent of escalate the access privilege of TAs' to that of Bob (they can modify the executable and make it do anything on the files Bob has access to). The exercise could be solved by read, write, execute permissions.