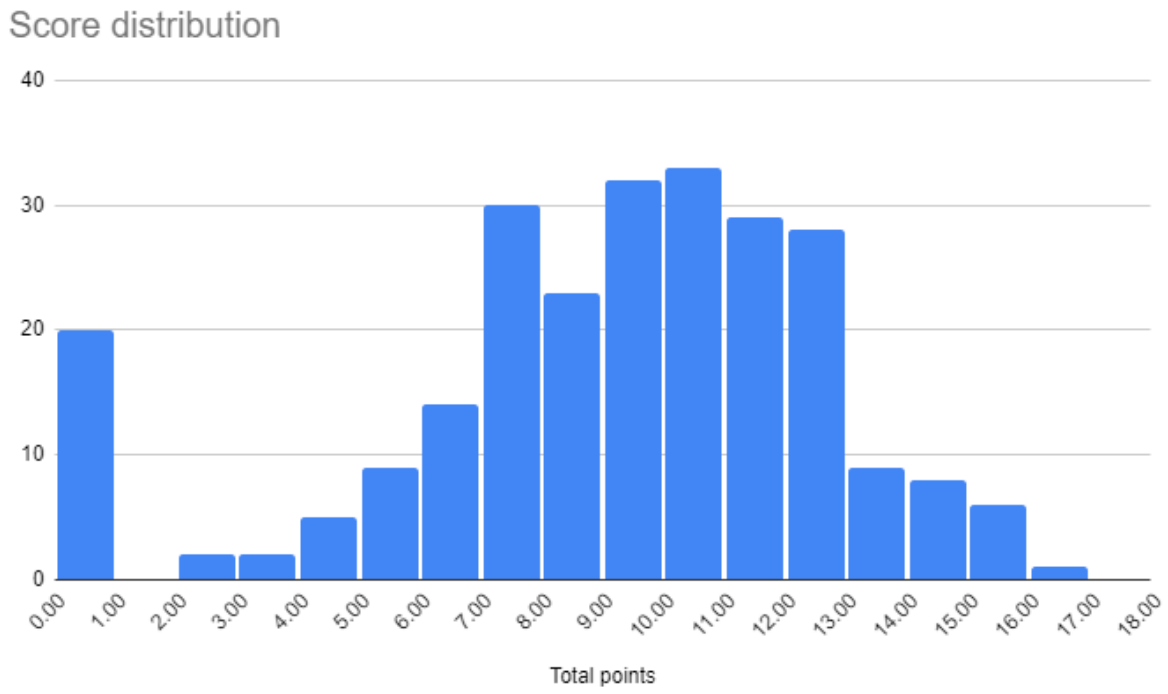
Most Repeated Errors- Midterm 2023



General notes about the midterm results

- We report a score out of the maximum total number of points, not a grade on a scale between 1 and 6.
- It will only be reflected in the final grade (30%) if it improves your grade, if it lowers it we will only take the grade you obtain in the final exam.
- The average number of points was 9.71, the median 9.25; one student has 16 points, and the maximum is 18.
- Students that did not participate in the midterm (which is ok since it can only help), are included in the figure for transparency. Their scores were not accounted for when computing the mean.

The results are in line with the experience from prior years: the midterm serves as a checkpoint for students to realise whether they are understanding the course, and for some students to realise that attending or watching lectures is not sufficient to solve the questions in the exam, and that exercise sessions can really help.

General Advice

- Answer *all parts* of a question. For example:

    - If the question asks about principles, use the principles.
    - If the question asks for the name of a property, provide the name of a property.
    - If the question asks for the application of a concept, describe how you would apply the concept. Just paraphrasing the lecture or the definition without stating how it relates to the scenario in the question results in no points, as we cannot assess your understanding.
    - If the question asks to justify, justify (providing just names without justifications rarely gives points). Answering partially, results in partial points.
- Answer the question we ask, and try to keep your answer as *concise* as possible. Wrong statements about aspects that do not appear in the question (e.g. a wrong countermeasure when the question does not ask for a countermeasure) can lead to deduction of points.
- Do not exceed the space limit. The questions are designed such that the space is sufficient for the answer. We don't grade parts of the answer beyond the 5 lines allowed. Writing more lines in small font not only makes the answer unreadable but is also an abuse that creates unfairness among students. Please be respectful of the rules and your peers.

MCQ

Q1

A confusion is not always a confused deputy problem. The term "confused deputy" is a fixed security terminology which describes a precise event in which an entity without privileges to perform an action is able to make another entity with higher privileges to perform the action. The fact that the maintenance team believes there is a leak can be confusing, but in itself is not a security problem. The confused deputy problem happens when the confused deputy (the IT maintenance team) book the room (make an action with higher privileges).

Q6

Encrypt-and-Mac ensures integrity of the exchange. A number of students marked (A) Enc(K_1, m), MAC(K_1, m) or (D) Enc(K_1, m), MAC(K_2, m) as the option that would guarantee "confidentiality and integrity of the exchange". As explained in the lecture, those options would guarantee integrity of the plaintext m, but not of the whole exchange (in particular it does not guarantee Enc(K,m) has not been tampered with.
In order to have integrity of the exchange, you need to use the Enc-then-MAC construction in which the MAC is computed on Enc(K,m) and therefore ensures the integrity of the ciphertext that is being sent.

Q7

The seed of a token-based authentication does not need to be secret. In a token authentication mechanism, the seed is an agreed value between token and server used to bootstrap the process. It is necessary that they agree on this value for the protocol to work (so that they compute on the same inputs), but the value does not need to be secret.

Think about the whole protocol in which the value computed at step n, which will be the seed of round n+1 is sent over the wire. What makes the protocol secure is not that this value is secret, is that only the token having the shared key with the server can perform the correct computation on this value.

Open Questions

Fast Thesis Pitch

Part 1:
Not reasoning about the security policy and mechanisms when reasoning about psychological acceptability. Several answers reasoned that psychological acceptability is followed because the interface is simple, or authentication is not required. But psychological acceptability is not about the simplicity of the system or its functionality, but about how well users can understand the security policy and the security mechanisms. In the voting application there are no security checks, as anyone can vote any number of times, so there cannot be psychological acceptability.

Stating that least privilege is followed because users can only access the excel sheet through the form and cannot read the result from the sheet. But this ignores the fact that users can write to the sheet as many times as they want. Restricting some privileges doesn't imply that the least amount of privileges are given and hence does not imply that the least privileges principle is followed.

Part 2:
Stating that applying separation of privilege solves the problem without a correct justification. Several answers stated that applying separation of privilege i.e. multiple users required to vote once, or applying 2FA solves the problem of boosting votes. Neither approach guarantees that users can’t vote multiple times for the same presenter.

Confusing separation of privilege with least privilege. Stating that separation of privilege is not followed because users have more privileges than necessary i.e. users don't have least privileges. Enforcing separation of privileges does reduce one entity's privilege, but it doesn't say anything about whether the least privilege principle is applied or not. The difference between the two principles must be clear.

Interviewing Journalists

Including organisations into the conflict set. The Chinese Wall model has a goal to avoid interactions between entities that have a “conflict of interest”. In the problem description, it is explicit that the publications are competing against each other, and hence, have conflict of interest. It is also explicit that the organisations do not compete against each other, and hence, members of organizations do not have conflict of interest. The conflict set in the answer should only include the publications.

Missing the historical employment when putting journalists into groups. One of the three basic concepts of the Chinese Wall model, as shown in the lecture slides, is “Subjects are associated with a history of their access to objects, and in particular their labels.” Some answers did not take into account historical employment of journalists resulting in groups including journalists having a conflict of interest, e.g., putting J3 (previously worked for Zurich Times) with J2 or J5 (working for Lausanne Journal).

Mis-reading the descriptions and mixing the names of journalists. Some answers correctly explain the Chinese Wall model, however, they make errors when defining the groups. In particular, several answers treat J3 as if it was J4. This unfortunate mistake results in point reduction as the answer is not correct. To avoid such mis-reading mistakes, please read the description carefully.

Adding arbitrary assumptions into the scenario. In the second part of this question, it is written in the problem description “Assign the **minimal** permissions that are **needed for this scenario**”. Some answers added assumptions that are not acceptable. It is ok to add assumptions if there is an underspecified part of the question needed for your answer, but you cannot add assumptions that change or extend the scenario, and thus change the question and its answer. We only grade the answer to our question, not to modifications made by students.

One example of such assumptions is “the researchers need to do research so they need read permission to everything”. In the above problem description, it says “for this scenario”, there is nothing about data processing for further research in the description. We note that, in reality, data processing is a separate part from data collection, and usually happens after the check (and consent) from the data subject. Thus, the permissions for data processing can be changed after this check, and can be completely independent from the permissions required in the collection scenario described in the question.

Another example of such assumption is “Nova creates these files so Nova has read/write/execution permissions”. Who created the files and when is irrelevant, as the permissions can change afterwards. The question sets up a scenario, in which the files are already there and asks for permissions at that moment.

There are many other assumptions that change the question, e.g., claiming there are only two groups which are `Researcher` and `GroupJ`, which does not appear in the question description.

Permissions added due to assumptions that changed the question, made the answers non-minimal with respect to the scenario in the question, and hence, resulted in points deduction.

Using the sticky bit. Some answers added the sticky bit to the permissions. The sticky bit, when set on a file, ensures that a file is kept in memory even when not in use (obsolete nowadays). When set on a directory, the files on this directory can only be deleted by the owner or root. None of these functionalities is needed in the scenario in the question.

Thinking that giving execution rights on `difference-audio-text.sh` to GroupJ suffices to run the script.
The `difference-audio-text.sh` script takes as input the audio file and the transcript file. Since the audio file is owned by `nova` and assigned to group `Researchers`, journalists `Jx` cannot access it in a restricted way (other than giving access to everyone in the system, which is not minimal). Thus, even if `GroupJ` has execute permission, the script cannot run. The only users that could have access to the audio file are `nova` (as owner) and members of group `Researchers`. Then, to allow `Jx` in `GroupJ` to run the script it would be needed to do this using Nova's permissions via the suid bit, and giving `nova` read permissions on the audio and the transcription text file.

Putting execution permission to files that are only used as input. Many answers set the execution bit to text or audio files, which in the question were explicitly described as being only an input. To be used as an input, files only need the read permission. By setting the execution bit, the permissions are not minimal.

Remark: misplacing the permissions or assigning them using binary values instead of r/w/x(s) is not correct. The UNIX permissions of a file are grouped always in the order of read, write, execute. Some answers mistakenly put the execute permission at the read permission position. Some answers used 1 or 0 in the table. This is not only a bad practice, but in this case hindered correction impossible as it does not allow to show whether special bits are used. We did not deduct points in these cases, but we would like to raise the point so that in the future everyone uses the standard UNIX convention that you have learned in the class and in the homework.

Fred Discussion

Define/Describe a property rather than justify why it is needed: Giving the definition of a hash function property is not a justification for why it is needed in the question scenario. A justification should give a clear security argument such as "To ensure that the adversary cannot do X, the hash function needs to have property Y."

Design a system rather than answer the question: The question did not ask whether the hashes of a post and its edit would be enough to prove an edit but to define the properties the hash function at least must have if they are to be useful for that. This means that no points were awarded to answers that instead of answering the question suggested additions or modifications to the system, such as digital signatures, symmetric cryptography, etc. to improve it.

Invent new hash function properties. The three relevant security properties of cryptographic hash functions are pre-image resistance, second pre-image resistance, and collision resistance. We did award points to answers that correctly described those properties without naming them. We did not give any points to answers that invented new hash function properties such as "non-repudiation" (a security property but not a property of a hash function).

State the same threat model in both question parts. To correctly answer the question, it was necessary to correctly model the adversary and its capabilities. Many stated that the threat model was the same in both parts of the question which is not correct.

Secure Accounting

Part 1:

Incorrect assumptions about k1.

A common error is to assume that k1 is shared with the server, when the question does not say that has happened (as opposed to k2). Adding your own assumptions about k1 changes the scenario you were presented with, and therefore the answer is not anymore valid.

Treating MACs and Signatures as encryption.

One cannot “decrypt the signature/MAC”. A signature is only used to validate the authenticity of a message: it supports two functions “Sign” and “Verify”. A MAC has similar functions. On top, depending on the signature/MAC mechanism chosen, it is not possible to recover the message.

Brute forcing on signature.

First, the large majority of signature schemes make it impossible to recover the message. Even if it was possible, recovering the message would not allow to verify the integrity of the messages nor protect against duplicate attacks (anyone can sign with the public key of the server).

Jumping to integrity and authenticity before decrypting

Many errors involve talking about integrity and authenticity without acknowledging that we cannot even decrypt the value L. The question asked whether the server can compute the total number of lollipops, not if it can be sure of its integrity.

Using K2 to decrypt Enc(K1, L).

Many mentioned that we can decrypt Enc(K1, L) with K2. The question states that Enc() is symmetric encryption. Therefore, we need the same key to encrypt and decrypt.

Avoiding to answer the question and contradicting statements.

Saying “In theory yes but …”, “It is hard to tell …”, and “Assuming we decrypt L, we get the sum of L”... only makes it seem that you are avoiding the question. When asked a direct yes/no question, answer directly and clearly. Any reasoning that does not conclude in a clear final answer results in points deduction.

Part 2:

Focussing only on the signature. Many of you highlighted the fact that the signature was produced using the public key of the server, and used this fact to argue that no, there is no way for the server to tell whether an LAYD app produced the message or anyone else. While this fact is true, the message also contains MACs. A common error was to not discuss that one of these MACs allows the server to be sure that the message comes from genuine LAYD apps..

When arguing against origin authentication, propose a fix/other solution rather than a security argument. Answers that consisted of proposing an alternative system such as "Instead the signature should be with the private key." received no points. The question asked about some properties of the stated message. Writing about another system does not answer the question we are asking.

When arguing for origin authentication, saying what honest participants (i.e., app and server) know and can do, but not the adversaries. Using statements such as "The app knows k1" or "The app can produce the signature on L" are most often necessary to justify your answer, but to make your answer complete, you also need to discuss what an adversary *cannot* do. It is important since saying the app can produce a signature (for example) is not sufficient to argue for integrity if an adversary can also produce the signature.

Part 3:

Not answering the question.

The questions ask about the influence of **Skey of the server** on proving to a **third party** the authenticity of the message.

Examples of avoiding the question:

- Suggesting a new mechanism. Although it might be correct, you don't answer the original question.
- Answering the question: Can the server verify the integrity of the message? By talking about k1 and k2 … Although some statements might be correct, they don't address the question.

Conflicting answer and justification.

Many start by answering yes, but add many conflicting statements to argue **for** and **against** at the same time without a conclusion at the end. The justification as to why it **DOES** or **DOES NOT** depend on **Skey** should flow from arguments about the current mechanism and why a third party can/cannot use it to verify. Depending on the coherence of the justifications, grading varied from deducting points to no points at all.

Mentioning that signatures are signed with the public key and verified with the private key.

This is plain wrong. By definition, it is exactly the opposite.

Blindly trusting the server and using MAC, k1, and k2.

Many answers point that the server can check using k2 and k1, and can confirm to the third party themselves. The server cannot use a MAC to **prove** to a third party they are not the sender of the message, as they know k2 and therefore could have created the messages themselves. As explained in the lectures, a symmetric key primitive cannot be used to prove message origin. Any answer that confuses a MAC with a signature was considered incorrect and points were deducted accordingly.

FELP

Claiming that one can brute-force biometrics.

Many answers claimed that biometrics is hard/impossible to brute force, without any definition of brute-force attack on Face Recognition. From the Security point of view, there are multiple ways to define this type of attack. We can define it as an offline attack where we aim to invert the stored template or an online attack where the goal is to generate a plausible face. Both are feasible, but there is no evidence that they are harder (or easier) than brute-forcing a password. Some answers tried to define brute forcing (even roughly) and explain why it is harder (for example comparing the dimensionality of inputs). Even though the dimensional point of view is arguable, since not any combination of pixels is a facial image and the adversary can generate valid facial images for example via StyleGAN, we awarded full points to answers *supported by arguments* with full point.

Making an argument about natural problems with availability, not security issues.
Some answers described as a disadvantage that the face of a person may change with time and this person will lose access to the service. This argument is about loss of availability and does not consider any adversarial behaviour. The fact that a person may need to update their photo every few years is not a degradation of security. We would have accepted this argument if there was adversarial behaviour, like an adversary forcing changes in the face of someone (e.g., throwing acid to their face), causing denial of service to this person. We did not encounter such arguments in the answers.

Answering using contradictory assumptions.
Many answers did not clearly formulate assumptions about adversarial capabilities, and the answers themselves often implied contradicting assumptions. For example, some answers claimed that an advantage of the biometrics-based system is that it makes it impossible to impersonate another user; and that a disadvantage is that photo spoofing allows for impersonation. If you assume that you have a biometric system which detects spoofing, then the advantage holds, but the disadvantage is wrong. On the other side, if photo spoofing is possible, then impersonation is also possible and the stated advantage is incorrect.
In order to avoid such errors, please reflect on the adversarial capabilities that you are using in your answers and do not change them in between subquestions.