Security Handin2

Mads Christian Nørklit Jensen

October 2024

1 GDPR & medical data as plain text

Health data refers to personal information, that relates to the health status of a person. This data is considered sensitive data, which in turn is defined as data that reveals things such as race, political opinions, beliefs, but most importantly in this scenario, bio-metric data for the purpose of uniquely identifying a natural person or data concerning health.[1][2] Furthermore it is also stated that one should not process more personal data than necessary, as well as organisations being required to carry out risk assessment and develop, where necessary, specific security measures on access control and management of information on health data.

In our scenario several issues arise if the hospital simply stores the patients private data. Fir this scenario we are only interested in the private value they input, so if we store other information, such as their names, we are violating the rule of not processing more personal data than necessary. Whilst this specific issue is solved by only storing the information that will be used, several risks are still present. First and foremost, the data has to be conveyed from the patient to the hospital, which in this case is done over a central server the hospital runs. If the network is unsafe, for example in case of a Dolev-Yao attacker, unless security measures are taken, the data processed through the network is at risk of being accessed by unauthorized parties, this could be malicious attackers, but also simply other patients. In addition, if the hospital simply stores the patients private input, the data is again at risk unless security measures are taken.

To combat some of these issues, the hospital has been suggested to use a Federated Learning algorithm that supports Secure Aggregation, such that the values being sent on the network and accessed by the model is only aggregate values, and not the patients' individual private plain text data. Whilst this approach does shore up some of the risks, this approach is still flawed, as the network is still unsafe. The data transmitted could be intercepted or manipulated, a malicious attacker might even inject faulty data into the system. To truly secure the hospital and the patients', a combination of Federated Learning with Secure Aggregation - to protect the sensitive data, and TLS combined with certificates - to provide secure communication across the network, is needed.

2 Adversarial model, proposed solution and why it provides security

Adversarial Model

For this scenario we face two adversaries, the patients which can be classified as passive adversaries as they are actively listening on the network, whilst not interfering with any data or performing malicious attacks. The second adversary is a Dolev-Yao attacker, an attacker assumed to have full control over the communication network. In this model, a Dolev-Yao attacker can intercept, modify, delete and inject messages into the network. The attacker can also impersonate legitimate people on the network, by crafting fake messages and can even combine intercepted parts of different messages, to send new messages that follow certain standards.

Secret Sharing & Aggregation Protocol

To ensure that patients and the hospital never has access to patient's private data, I've implemented n-out-of-n Additive Secret Sharing. Each patient inserts their private input, and divides it into three shares (since we in this scenario have 3 patients in total). The algorithm then generates n-1 random shares, so in this case, 2 random shares for each patient. The final share is then computed as $s_n = s - (s_1 + s_2 + \ldots + s_{n-1})$ where s is the patients private input (the secret). The secret can then only be reconstructed when all n shares are available. If any share is missing, the secret will remain hidden, ensuring the security. This aggregated value is then sent to the hospital. The hospital collects all aggregated shares and computes the final sum, which is the aggregate of all the patients private input. The hospital is therefore only ever in contact with aggregated values, ensuring that GDPR laws are upheld for all sensitive data.

2.1 TLS & Certificates

To ensure secure communication on the network, to protect against a Dolev-Yao attacker, I've implemented certificates and TLS. First I run a script where a root Certificate Authority is created, which is used to sign server and client certificates. A seperate key and Certificate Signing Request is then generated for the hospital as well as for each patient. The CA signs these request to generate certificates for both patients and the server. Since patients will be communicating both with each other and the server, I've implemented mutual TLS, so both the server and patents can authenticate against a trusted Certificate Authority, to ensure they are indeed communicating with the intended other party (server & patient). The Certificate Authority ensures that only legitimate entities are issued a certificate, preventing impersonation attacks.
TLS uses symmetric and asymmetric encryption to secure data in transit. After the initial handshake to establish connection, a shared secret is established,

which is used to decrypt the messages, ensuring that even if a message is intercepted, it cannot be read. TLS also uses exchange algorithms to establish it's session keys securely. An attacker will have to solve the mathematical problems, which currently is infeasible.

References

[1] Health data in the workplace. https://www.edps.europa.eu/data-protection/data-protection/reference-library/health-data-workplace_en.

[2] S' of data protection. https://www.edps.europa.eu/data-protection/data-protection/glossary/s_en#sensitive_data.