Homework: Sign and Encrypt

Jackson Mowry

Thu Nov 6 09:36:03 2025

The Benefits of an Encrypt Then Authenticate Protocol in Secure Communication

When implementing secure communication in our upcoming project it is important that we learn from past cryptographic research in order to not find ourselves in any common pitfalls. Users of such systems have expectations that the platform being utilized supply both authentication and encryption to ensure that messages received both come from the indicated party, and have not been tampered with in transit. It is also important that burden of verifying both properties of a message lie mostly on automated systems in the platform, with easy ways for the user to verify correctness. I propose that the next upgrade to our platform moves to using an Encrypt Then Authenticate protocol, as well as providing simple reauthentication methods between parties.

Modern users expect privacy when using a system, which for the most part boils down to concepts such as end-to-end encryption and authentication. While a naive approach to encryption can "check-off" one of those requirements, it comes with many potential downsides. Sign and Encrypt is a general classification of algorithms which first "sign" the contents (i.e. generating a MAC), and then encrypting the entire package before sending it to the recipient. One major issue with this strategy is that when unencrypted the contents (sent from A -> B) of the message have already had their MAC generated, meaning that package can then be forwarded to another user effectively allowing user B to "forward" a message directly from user A to user C.To avoid this potential pitfall a potential solution is to ensure that the intended recipient is included as part of message body, meaning that the generated MAC will change if the field is modified.

While this solves the original problem it can still leave a system vulnerable to a potential failure case. Due to the MAC itself being encrypted, the "package" is required to be entirely decrypted before use. This fact opens our system to potential vulnerabilities dealing with the integrity of the message. If an attacker can intercept the message and modify it there is potential for revealing secret

information to the attacker. To mitigate this potential issue we should ensure that our algorithm follows the Encrypt Then Authenticate pattern. This pattern ensures that any modifications to the encrypted package are instantly caught when comparing the MACs, leading to a safer system that does not attempt to operate on compromised data.

While the proposed Encrypt Then Authenticate protocol gains us the above described property, it is potentially a harder challenge to implement. as we must be care when choosing what ends up in the encrypted "package". If we forget to include various values required for decryption in the MAC they can be changed, leading us right back into the original pitfall described where a message could be modified, leading to invalid cipertext.

Lastly, we come to the topic of easing the burden of verification from our users. While communication platforms often provide the cybersecurity features mentioned above, they often lack to provide quick authentication between users. This feature could take the form of a feature that allow users to issue challenges to the other party that verify they are who they say they are, without having to reveal potentially secret information.

In conslusion, the benefits of an Encrypt Then Authenticate protocol fulfil many of the desires of a user interested in their own security and privacy, and therefore should be implemented in our next platform. Additionally, our platform should provide and easy method to initiate reauthentication between users to verify their communication channel has not been compromised over time.